Recuperador de Portadora de un Receptor Digital con Modulación QAM implementado en un DSP

Que para obtener el Título de

'Igneriero en Telemática'

Presenta

Gerardo Zamitis Ortega

Asesores

Dr. Aldo G. Orozco Lugo
Ing. César I. Pérez Macías
Ing. Fernando Téllez Alcaráz

México D.F. Junio del 2005
ÍNDICE DE FIGURAS ........................................................................................................... iii
ÍNDICE DE TABLAS ............................................................................................................... v
RESUMEN ............................................................................................................................. vi
OBJETIVOS ........................................................................................................................... vii

INTRODUCCIÓN ...................................................................................................................... 1
1. Antecedentes ....................................................................................................................... 3
  1.1 Sistema de Comunicaciones Digital Básico ................................................................. 3
  1.2 Transmisor .................................................................................................................... 4
  1.2.1 Modulación Digital .................................................................................................. 4
  1.2.2 Modulación en Amplitud por Cuadratura (QAM) ..................................................... 4
  1.3 Receptor ....................................................................................................................... 5
  1.3.1 Demodulación Digital Pasa Banda .......................................................................... 5
  1.4 Recuperación de Portadora ........................................................................................... 6
    1.4.1 Panorama General de los Recuperadores de Frecuencia de la Portadora .......... 7
    1.4.2 Algoritmos de Estimación de la Frecuencia con Datos Auxiliares ..................... 8
      1.4.2.1 Método Kay ..................................................................................................... 8
      1.4.2.2 Método Fitz .................................................................................................... 8
      1.4.2.3 Método Luise & Reggianiini .......................................................................... 9
      1.4.2.4 Método de la Aproximación de la Estimación de la Máxima Similitud ........ 9
      1.4.2.5 Comparaciones de Funcionamiento ............................................................... 10
    1.4.3 Algoritmo de Estimación sin datos auxiliares pero con recuperador de reloj auxiliar
          1.4.3.1 Algoritmo de lazo Cerrado ............................................................................ 10
    1.5 Recuperación de la Fase de la Portadora .................................................................... 11

2. Planteamiento del Problema ............................................................................................ 12
  2.1 Algoritmo de Retardo y Multiplicación sin Información ........................................... 13
  2.2 Algoritmo de Lazo Abierto con Información de Temporización ............................... 16
  2.3 Estimación adelantada con QAM (Recuperador de Fase) ........................................ 18
  2.4 Solución Propuesta ..................................................................................................... 19

3. Desarrollo de la Propuesta ............................................................................................... 20
  3.1 Diagramas de Flujo ........................................................................................................ 21
  3.2 Simulaciones de los distintos algoritmos a implementar .......................................... 24
    3.2.1 Simulaciones del algoritmo "Retardo y Multiplicación sin Información" ............ 24
    3.2.2 Simulaciones del algoritmo "Retardo y Multiplicación con Información de
           Temporización" ........................................................................................................ 30
    3.2.3 Simulaciones del algoritmo "Estimación Adelantada con QAM" ......................... 36
    3.2.4 MATLAB ............................................................................................................. 39
  3.3 Introducción a los DSPs ............................................................................................... 40
    3.3.1 ¿Qué es un DSP? .................................................................................................. 40
    3.3.2 ¿Por qué un DSP? ............................................................................................... 40
  3.4 DSK o DSP Starter Kit ................................................................................................. 41
  3.5 Code Composer Studio ............................................................................................... 41
3.6 DSPTMS320F2812 ................................................................. 42
3.6.1 Arquitectura del DSP TMS320F2812 .................................. 43
3.6.2 Mapa de Memoria .............................................................. 44
3.6.3 IQmath, máquina virtual flotante ........................................ 46
3.6.4 SCI, Interfaz de Comunicación Serial .................................. 47
3.6.5 Algoritmos en el DSP ........................................................... 47
3.7 Interfaz de comunicación serial DSP - MATLAB .................. 47
3.7.1 Interfaz de comunicación serial del DSP (SCI, Serial Communications Interface) .... 47
3.7.2 Diagramas de flujo de la comunicación DSP - MATLAB .......... 49
3.8 Diseño y Fabricación de la tarjeta independiente .................. 51
3.8.1 Circuito Regulador de 5 Volts .......................................... 51
3.8.2 Circuito Regulador de 3.3 Volts ....................................... 51
3.8.3 Circuito Acondicionador de señal ...................................... 52
3.8.4 Circuito Regulador de 1.9 Volts ....................................... 52
3.8.5 Circuito de Reloj .............................................................. 53
3.8.6 Circuito MAX232 ............................................................. 53
3.8.7 Circuito del DSP .............................................................. 54
3.8.8 Tarjeta Electrónica impresa ................................................. 55

4. Validación del Sistema ............................................................... 57
4.1 Resultados obtenidos del DSP TMS320F2812 con CCS ...................... 57
4.2 Resultados obtenidos del DSP TMS320F2812 con MATLAB ............... 61

5. Conclusiones ......................................................................... 62
5.1 Conclusiones ................................................................. 62
5.1 Mejoras ........................................................................ 63

REFERENCIAS ........................................................................... 65

BIBLIOGRAFÍA .......................................................................... 66

GLOSARIO .................................................................................. 67

APÉNDICE A .............................................................................. 68
ÍNDICE DE FIGURAS

1. Antecedentes .................................................................................................................. 3
1.1 Diagrama de bloques esenciales de un Sistema de Comunicaciones Digitales........... 3
1.2 Modulador QAM........................................................................................................... 5
1.3 Ejemplos de Constelaciones QAM.............................................................................. 5
1.4 Diagrama a bloques del Estimador de Frecuencia L&R ........................................ 9
1.5 Diagrama a bloques de la generación del error en frecuencia .................................. 10

2. Planteamiento del Problema .......................................................................................... 12
2.1 Transmisor del DCS .................................................................................................. 12
2.2 Receptor del DCS...................................................................................................... 12
2.3 Esquema de Retardo y Multiplicación sin Información ........................................... 13
2.4 Diagrama a bloques para el estimador digital ............................................................ 14
2.5 Varianza del error del estimador Retardo-Multiplicación sin Información............... 15
2.6 Diagrama a bloques de un estimador de lazo abierto ............................................. 16
2.7 Varianza de la Frecuencia Normalizada en función de la Es/No ............................... 17
2.8 Diagrama a bloques del Recuperador de Fase ......................................................... 19
2.9 Desempeño del Estimador de Fase ............................................................................ 19

3. Desarrollo de la propuesta ............................................................................................ 20
3.1 Diagrama de flujo del algoritmo "Retardo y Multiplicación sin Información" ......... 21
3.2 Diagrama de flujo del algoritmo "Retardo y Multiplicación con Información de temporización" .......................................................... 22
3.3 Diagrama de Flujo del algoritmo "Estimación Adelantada con QAM" ..................... 23
3.4 Varianza del Error del Estimador "Retardo Multiplicación sin Información con 100 símbolos y con exceso de ancho de banda de 0.25" ........................................ 24
3.5 Varianza del Error del Estimador "Retardo Multiplicación sin Información con 150 símbolos y con exceso de ancho de banda de 0.25" ........................................ 25
3.6 Varianza del Error del Estimador "Retardo Multiplicación sin Información con 200 símbolos y con exceso de ancho de banda de 0.25" ........................................ 25
3.7 Varianza del Error del Estimador "Retardo Multiplicación sin Información con 300 símbolos y con exceso de ancho de banda de 0.25" ........................................ 25
3.8 Varianza del Error del Estimador "Retardo Multiplicación sin Información con 100 símbolos y con exceso de ancho de banda de 0.5" .......................................... 26
3.9 Varianza del Error del Estimador "Retardo Multiplicación sin Información con 150 símbolos y con exceso de ancho de banda de 0.5" .......................................... 26
3.10 Varianza del Error del Estimador "Retardo Multiplicación sin Información con 200 símbolos y con exceso de ancho de banda de 0.5" ......................................... 27
3.11 Varianza del Error del Estimador "Retardo Multiplicación sin Información con 300 símbolos y con exceso de ancho de banda de 0.5" ......................................... 27
3.12 Varianza del Error del Estimador "Retardo Multiplicación sin Información con 100 símbolos y con exceso de ancho de banda de 0.75" .......................................... 27
3.13 Varianza del Error del Estimador "Retardo Multiplicación sin Información con 150 símbolos y con exceso de ancho de banda de 0.75" ...................................... 28
3.14 Varianza del Error del Estimador "Retardo Multiplicación sin Información con 200 símbolos y con exceso de ancho de banda de 0.75" ...................................... 28
3.15 Varianza del Error del Estimador "Retardo Multiplicación sin Información con 300 símbolos y con exceso de ancho de banda de 0.75" ......................................................... 28
3.16 Salida del algoritmo Retardo-Multiplicación sin Información ........................................ 29
3.17 Varianza del Error del Estimador "Retardo Multiplicación con I. de T. con 100 símbolos y con exceso de ancho de banda de 0.25" ................................................................. 30
3.18 Varianza del Error del Estimador "Retardo Multiplicación con I. de T. con 200 símbolos y con exceso de ancho de banda de 0.25" ................................................................. 31
3.19 Varianza del Error del Estimador "Retardo Multiplicación con I. de T. con 300 símbolos y con exceso de ancho de banda de 0.25" ................................................................. 31
3.20 Varianza del Error del Estimador "Retardo Multiplicación con I. de T. con 100 símbolos y con exceso de ancho de banda de 0.5" ................................................................. 31
3.21 Varianza del Error del Estimador "Retardo Multiplicación con I. de T. con 200 símbolos y con exceso de ancho de banda de 0.5" ................................................................. 32
3.22 Varianza del Error del Estimador "Retardo Multiplicación con I. de T. con 300 símbolos y con exceso de ancho de banda de 0.5" ................................................................. 32
3.23 Varianza del Error del Estimador "Retardo Multiplicación con I. de T. con 100 símbolos y con exceso de ancho de banda de 0.75" ................................................................. 32
3.24 Varianza del Error del Estimador "Retardo Multiplicación con I. de T. con 200 símbolos y con exceso de ancho de banda de 0.75" ................................................................. 33
3.25 Varianza del Error del Estimador "Retardo Multiplicación con I. de T. con 300 símbolos y con exceso de ancho de banda de 0.75" ................................................................. 33
3.26 Varianza del Error del Estimador "Retardo Multiplicación con I. de T. con 100 símbolos y con exceso de ancho de banda de 1.0" ................................................................. 33
3.27 Varianza del Error del Estimador "Retardo Multiplicación con I. de T. con 200 símbolos y con exceso de ancho de banda de 1.0" ................................................................. 34
3.28 Varianza del Error del Estimador "Retardo Multiplicación con I. de T. con 300 símbolos y con exceso de ancho de banda de 1.0" ................................................................. 34
3.29 Salida del algoritmo Retardo-Multiplicación con Información de Temporización ............... 35
3.30 Varianza del Error en fase del Estimador en Fase para 100 símbolos .......................... 36
3.31 Varianza del Error en fase del Estimador en Fase para 200 símbolos .......................... 37
3.32 Varianza del Error en fase del Estimador en Fase para 300 símbolos .......................... 37
3.33 Varianza del Error en fase del Estimador en Fase para 400 símbolos .......................... 37
3.34 Varianza del Error en fase del Estimador en Fase para 500 símbolos .......................... 38
3.35 Varianza del Error en fase del Estimador en Fase para 1000 símbolos ......................... 38
3.36 Recuperación de Fase de la señal portadora ............................................................ 39
3.37 DSP Starter Kit TMS320F2812 eZdsp ............................................................... 41
3.38 Presentación de Code Composer Studio ............................................................. 41
3.39 Arquitectura del DSP TMS320F2812 ........................................................................ 43
3.40 Mapa de Memorias del DSP TMS320F2812 ......................................................... 44
3.41 Modo de transmisión de Datos por el SCI ......................................................... 48
3.42 Diagrama de Flujo de la comunicación serial en el DSP ........................................... 49
3.43 Diagrama de Flujo de la comunicación serial en MATLAB .................................... 50
3.44 Circuito Regulador de 5 Volts ........................................................................ 51
3.45 Circuito Regulador de 3.3 Volts ........................................................................ 51
3.46 Circuito Acondicionador de Señal ........................................................................ 52
3.47 Circuito Regulador de 1.9 Volts ........................................................................ 52
3.48 Circuito de Reloj ............................................................................................... 53
3.49 Circuito MAX232 ............................................................................................. 53
3.50 Circuito del DSP ............................................................................................... 54
3.51 Diagrama a bloques del sistema general de la Tarjeta Independiente ........................................54
3.52 Cara superior del circuito impreso ..........................................................................................55
3.53 Cara inferior del circuito impreso ...........................................................................................55
3.54 Cara superior del circuito ........................................................................................................56
3.55 Cara inferior del circuito ........................................................................................................56

4. Validación del Sistema ............................................................................................................57
  4.1 Fotografía del DSK TMS320F2812 ........................................................................................57
  4.2 Salida del algoritmo "Retardo-Multiplicación sin Información" en CCS ........................58
  4.3 Salida del algoritmo "Retardo-Multiplicación con I. de T." en CCS .................................59
  4.4 Salida del algoritmo "Estimación Adelantada con QAM" .....................................................60
  4.5 Interfaz serial de la PC con el programa MATLAB .............................................................62

ÍNDICE DE TABLAS

3. Desarrollo de la propuesta ........................................................................................................20
  3.1 Características Generales del TMS320F2812 .................................................................42
  3.2 Tipos de Datos, Resolución y Precisión IQmath ..........................................................41

4. Validación del Sistema ............................................................................................................57
  4.1 Tabla de Funcionamiento de la interfaz serial DSP-MATLAB ......................................61
Recuperador de Portadora de un Receptor Digital con Modulación QAM implementado en un DSP.

**Palabras Clave:** Recuperador de Portadora, frecuencia, fase, QAM, DSP.

**Abstract:** This paper presents simulations of three algorithms, two of them are for Carrier Frequency Recovery and the third one is for a Carrier Phase Recovery. Those three algorithms were programmed in a DSP (Digital Signal Processor). The carrier recovery methods from above are appealing for burst mode transmissions.

**RESUMEN**

La recuperación de portadora tiene dos funciones generales: la primera de ellas es el de calcular un estimado del desplazamiento en frecuencia de la señal entrante, y la segunda es el de de-rotar la señal de entrada para compensar el defasamiento de la señal.

Existen distintos tipos de algoritmos de Recuperación de Portadora según su aplicación, una forma de caracterizarlos es la siguiente:

- Algoritmos clásicos. Hacen uso de todos los elementos necesarios para poder obtener el desplazamiento en frecuencia de la señal recibida, son casos ideales.
- Algoritmos con información. Hacen uso de la temporización o de datos auxiliares. Son reales, pero necesitan de aproximaciones previas o de información confiable.
- Algoritmos sin información: Sus características de funcionamiento son reales, y no hacen uso de algún tipo de información dentro del mismo receptor.

De las clasificaciones anteriores se eligieron como algoritmos de recuperación de portadora a los llamados "Retardo y Multiplicación" (Delay and Multiply) sin información y con información de temporización. Cabe mencionar que ambos algoritmos son de lazo abierto (esta configuración es para el modo de transmisión de ráfaga) y fueron escogidos debido a las características que debían cubrir estos módulos de acuerdo al sistema de comunicaciones que se está realizando en la sección de Comunicaciones del CINVESTAV. Así como también, se eligió un algoritmo de Recuperación de Fase llamado "Estimación Adelantada con QAM".

Se han simulado los algoritmos en MATLAB obteniendo resultados esperados, mostrados en el presente documento.

Los algoritmos han sido programados en la memoria Flash del DSP TMS320F2812 de Texas Instruments mediante su entorno de desarrollo llamado Code Composer Studio, para que el mismo pueda ser montado en un futuro sobre una tarjeta independiente.
OBJETIVOS

Objetivo General

• Diseñar y construir el módulo digital de recuperación de portadora de un receptor con demodulación QAM implementándolo en un DSP.

Objetivos Particulares

• Comprender el problema de Portadora.
• Revisar distintos algoritmos de Recuperación de Portadora.
• Elegir el algoritmo de acuerdo a las características del sistema de comunicaciones que esta siendo desarrollado en la sección de Comunicaciones del CINVESTAV.
• Simular y evaluar los algoritmos de Recuperación de Portadora escogidos en MATLAB.
• Aprender funcionamiento base del DSP TMS320F2812 de Texas Instruments.
• Utilizar herramientas de desarrollo del DSP TMS320F2812.
• Crear y Diseñar en Protel una tarjeta independiente para el módulo de Recuperación de Portadora.
INTRODUCCIÓN

El ser humano por naturaleza es un ser social y es esta naturaleza la que lo lleva a comunicarse con los demás. Hoy en día, la necesidad por comunicarse rebasa aquel tipo de comunicación que se hacía con gestos y sonidos, es llevada al punto de poder comunicarse con alguna otra persona del otro lado del mundo. Los sistemas de comunicación actual tratan de hacer esto cada vez más cotidiano para todos, mejorando la tecnología hasta llevarla siempre a nuevos horizontes que parecen ser imposibles de alcanzar, pero que en pocos meses se ven obsoletos bajo el ritmo en el que vive el mundo actual. Es una realidad, los sistemas de comunicación ya son hoy en día la infraestructura de toda la sociedad.

El presente Trabajo Terminal, hace uso de conocimiento existente en la literatura.

Los algoritmos de Recuperación de Portadora así como el de Recuperación de Fase son acordes a un sistema de comunicaciones con ciertas características. Es por lo anterior, que siendo parte de un sistema de comunicaciones (siendo más específicos, en el receptor) es necesario el llevar estos algoritmos a la práctica.

Los algoritmos de recuperación de portadora resuelven dos problemas generales: el primero de ellos es el del reloj con el que genera la señal portadora el transmisor, así como con el reloj que se cuenta en el receptor. Por cuestiones propiamente físicas, se sabe que nada es idéntico y es por lo anterior que existen desplazamientos en frecuencia simplemente por los dos relojes. El segundo punto corresponde al llamado efecto Doppler; se entiende como efecto Doppler cuando el frente de onda experimenta un efecto acordeón, es decir, puede existir expansión o comprensión del frente de onda, cuando se analiza un marco referencial de movimiento de dos objetos. Lo anterior, se puede entender más fácilmente con el siguiente ejemplo: Imagine que se encuentra en su auto, cuando de pronto escucha una sirena de ambulancia, el efecto que causa conforme se acerca a Usted es el de escuchar la sirena de forma más aguda, y conforme se aleja, escucha a la sirena mas ronca. Es exactamente el mismo efecto que sucede con la señal cuando llega al receptor, presenta desplazamientos en frecuencia y para un sistema de detección coherente como el que se esta implementando, es necesario un recuperador de portadora.

El presente Trabajo Terminal contiene cinco capítulos, los cuales serán descritos a continuación:

Capítulo 1.
Se aborda el tema de las comunicaciones digitales, los elementos que la conforman para dejar en claro el problema que se intenta resolver. Posteriormente se trata el Marco Teórico del problema de Recuperación de Portadora así como del problema de Recuperación de Fase.

Capítulo 2.
Se detalla el problema que origino al mismo proyecto. En el CINVESTAV existe la necesidad de realizar un Sistema de Comunicaciones Digitales, es así como el problema desea ser resuelto, a su vez se hace una descripción detalla de los algoritmo a implementar para así, al final de éste capítulo se de una solución al problema existente.
Capítulo 3.
Los algoritmos tratados en este Trabajo Terminal son detallados en este capítulo, mostrando los diagramas de flujo utilizados para la programación de las simulaciones realizadas, así como los resultados obtenidos de las mismas. Se abordan las distintas herramientas utilizadas para la construcción del prototipo, desde su descripción teórica hasta su aplicación.

Capítulo 4.
Se trata la validación de los módulos del Sistema Comunicaciones Digitales desarrollados en el presente Trabajo Terminal, en la tarjeta de desarrollo de Texas Instruments eZdsp TMS320F281. Teniendo como primera evaluación el desempeño de los algoritmos implementados en el DSP, y como segunda evaluación se tendrá el desempeño de funcionamiento de la interfaz serial DSP-MATLAB, para la transmisión de los resultados obtenidos de los algoritmos implementados sobre el DSP.

Capítulo 5.
Se exponen las conclusiones del presente Trabajo Terminal donde se da una breve explicación de lo comprendido así como de la importancia de lo desarrollado, teniendo como referencia tanto al objetivo general, así como de los particulares. Se da a conocer al lector, las ventajas y desventajas del Trabajo Terminal, así como de las posibles mejoras.
1. ANTECEDENTES

1.1 SISTEMA DE COMUNICACIONES DIGITALES BÁSICO

Los sistemas de comunicaciones, están compuestos muy a groso modo por 3 elementos: transmisor, canal y receptor; el trabajo actual será realizado en el receptor, desarrollando solamente un módulo.

La figura 1.1 muestra el flujo típico y esencial de una señal que pasa a través de un sistema de comunicaciones digitales (DCS; Digital Communication System), y que muy a groso modo muestra las tres partes en las cuales esta conformado: transmisor, canal y receptor [1].

La información fuente, es la información que se desea enviar por medio de un DCS. Después, el Codificador (o código fuente) transforma la información fuente en símbolos digitales; hace que la información sea compatible con el procesamiento de señales dentro de un DCS [1]. Después del Codificador, sigue la Modulación que es el proceso mediante el cual, los símbolos se convierten en formas de onda que son compatibles con el canal de transmisión, aquí hablamos de Modulación en Banda Base (cuando las formas de onda son pulsos) o Modulación Pasa Banda (la señal de información deseada modula una señal de alta frecuencia, llamada portadora). Después de la Modulación viene una etapa de acondicionamiento de la señal en el transmisor para poderla enviar a través de la antena hacia el canal de transmisión. El canal de transmisión es el medio físico en donde viaja la forma de onda e introduce una serie de errores a la señal. El receptor tiene una antena mediante la cual, captura la forma de onda enviada del transmisor, para después pasar a un circuito analógico acondicionador de la forma de onda, y así pasar al Demodulador que transforma la forma de onda (sea en Banda Base o en Pasa Banda) en los símbolos que fuesen antes modulados para su tratamiento digital. Los símbolos que provienen del Demodulador son tomados por el Decodificador del Receptor que transforma a los mismos en la información a como estaba en un inicio [1].

El párrafo anterior describe un funcionamiento utópico de un DCS, ya que no existen etapas como: codificación de canal, que es el añadir símbolos de información en la forma de onda, para que la probabilidad de que el ruido afecte a la información sea más pequeña; encriptación, que previene al sistema de comunicaciones de usuarios no deseados; y otras tantas etapas que a fin de cuenta son esenciales hoy en día para salvaguardar la
información en varios aspectos. Aclarando, que existen otras etapas y que el orden de las mismas bien podría ser distinto.

Cabe recalcar, que en un DCS el objetivo del receptor no es el de reproducir una forma de onda recibida con precisión; es, de hecho, el determinar de una señal afectada por el ruido, cual es la forma de onda de un conjunto finito de formas de ondas que han sido enviadas por el transmisor[1].

1.2 TRANSMISOR

1.2.1 Modulación Digital

La modulación digital es el proceso en el cual los símbolos digitales son transformados en formas de onda que son compatibles con las características del canal. En el caso de la representación en banda base estas formas de onda son pulsos, pero en el caso de la modulación pasa banda la señal de información modula una onda sinusoidal llamada Onda Portadora, o simplemente Portadora. Una razón por la cuál la modulación con portadora es un paso esencial para todos los sistemas que envuelven radio-transmisión, es por el tamaño de la antena. Lo anterior es, porqué la portadora es una señal sinusoidal de alta frecuencia; esto es funcional ya que la construcción de antenas tiene relación directa con la longitud de onda de la portadora. [1]

El desarrollo de las técnicas de modulación digital marcó las primeras etapas de digitalización de las redes de comunicaciones en el mundo. La modulación digital proporcionó un conjunto de métodos para convertir señales analógicas a cadenas de datos digitales y viceversa. El haberse trasladado a la modulación digital produjo una compatibilidad con servicios de datos digitales, una seguridad de datos más alta, una mejor calidad de comunicaciones, una disponibilidad de sistema más rápida, entre muchas otras características.

Debido a las características del presente Trabajo Terminal, la modulación pasa banda será la utilizada, más no desarrollada.

1.2.2 Modulación en Amplitud por Cuadratura (QAM)

Entre los formatos de modulación digital destaca el método conocido como QAM. La Modulación en Amplitud por Cuadratura (QAM, Quadrature Amplitud Modulation) es un esquema de modulación multi-nivel, con distintas combinaciones de amplitud y fase. Utilizando múltiples niveles, tanto en la modulación en amplitud como en la modulación en fase, es posible la transmisión de grupos de bits, de manera que cada uno de estos grupos será representativo de un conjunto nivel-fase característico de la portadora de la señal, mismo que dará cabida a un símbolo. Una de las características principales de la modulación QAM es que modula la mitad de los símbolos con una frecuencia y la otra mitad con la misma frecuencia, pero desfasada 90°. El resultado de las componentes después se suma, dando lugar a la señal QAM. Lo anterior se muestra en la siguiente figura.
De esta forma, QAM permite llevar dos canales en una misma frecuencia mediante la transmisión ortogonal de uno de ellos con relación al otro. La siguiente figura muestra constelaciones QAM [3].

1.3 RECEPTOR

1.3.1 Demodulación Digital en Pasa Banda

Cuando el receptor toma en cuenta la fase de la portadora para detectar las señales, el proceso se llama detección coherente; cuando el receptor no utiliza tal información de referencia de fase, el proceso se llama detección no coherente; por lo tanto, la estimación de la fase no es requerida.

El término demodulación y detección en las comunicaciones digitales son usados de forma indiferente, sin embargo, demodulación enfatiza el remover la portadora, y detección se refiere al proceso anterior, así como de la decisión del símbolo. En una detección coherente ideal, el receptor cuenta con todos los prototipos existentes para cada señal que llega al mismo receptor. Estos prototipos de forma de onda tratan de duplicar la señal transmitida en todos los aspectos. Es entonces cuando se dice que el receptor se amarra en fase con la señal entrante. Durante la detección, el receptor multiplica e integra (correlaciona) la señal entrante con cada una de sus réplicas [1].
La complejidad de un sistema no-coherente con respecto a un sistema coherente no es tanta, sin embargo, este beneficio trae consigo una deficiencia mayor, se paga con el factor de la probabilidad de error que será más grande en el primero.

La demodulación coherente es utilizada en las comunicaciones digitales pasa banda cuando un desempeño óptimo es de gran importancia. Lo anterior significa que la señal en banda base es derivada haciendo uso de una referencia local que posee la misma frecuencia y fase que la señal entrante. Esto requiere de una exactitud en la medición de la frecuencia así como de la fase, antes de que errores en fase introduzcan una interferencia entre los canales en cuadratura y en fase del receptor, degradando así el proceso de detección. El problema anterior es el tratado como la sincronía de portadora, o como algunos autores simplemente llaman, el problema de portadora.

La información de la fase de la portadora, debe hacerse mención, no es siempre necesaria. En aplicaciones donde la simplicidad y lo robusto de una implementación son más importantes que el lograr un desempeño óptimo, la demodulación no-coherente y coherente diferencial son técnicas alternativas para la detección coherente.

Es en esta etapa donde el Trabajo Terminal se ha enfocado, para un sistema de detección coherente, desarrollando el módulo de Recuperación de Portadora.

1.4 RECUPERACIÓN DE PORTADORA [2]

Primeramente se tiene que encontrar la frecuencia de la señal entrante.

Un sistema de recuperación de frecuencia logra 2 funciones básicas: (i) deriva una estimación $\theta$ del desplazamiento en frecuencia de la portadora.; (ii) de-rota la forma de onda recibida $r(t)$ a una velocidad angular de $2a\pi$; se refiere a derrotar la señal, cuando se desea que la velocidad angular sea cero y así los "puntos" que se observan en la constelación queden de forma estática. En la discusión que sobreviene distinguimos entre dos casos importantes:

(i) El desplazamiento es mucho más pequeño que $\frac{1}{T}$.

(ii) El desplazamiento está en el orden de la tasa de símbolos de $\frac{1}{T}$.

Donde $T$ es el periodo de la señal que se conoce que proviene del transmisor. El caso (i) ocurre cuando el receptor está funcionando en condiciones óptimas. En estas circunstancias, la información de la temporización puede ser recuperada primeramente, aún con la presencia de desplazamientos moderados en frecuencia, y entonces explotados para estimar $v$ [2]. El caso (ii) corresponde a la adquisición inicial de frecuencia en radios digitales de baja capacidad y de sistemas de comunicación satelitales. En estas aplicaciones nosotros podemos asumir que los símbolos de datos, la fase de la portadora, y quizás, la temporización sean todas ellas desconocidas [2].
Se pretende solucionar este problema, porque reduciendo el error de la frecuencia hacia un pequeño porcentaje de la tasa de símbolo es necesario antes de que alguna función de temporización pueda empezar exitosamente. Recordando que la etapa de temporización sería la más importante en un sistema receptor, para la obtención correcta de los bits o símbolos entrantes.

1.4.1 Panorama General de los Recuperadores de Frecuencia de la Portadora[2]

La recuperación de la frecuencia de la portadora con modulaciones lineales se puede clasificar en 2 ramas:

- Recuperadores de Frecuencia Analógicos.
- Recuperadores de Frecuencia Digitales.

Dentro de los mismos, existen varios métodos para recuperar la frecuencia de la portadora, los cuales se pueden clasificar de la siguiente forma:

- **Estimación de la Frecuencia con Datos Auxiliares.** De los recuperadores de frecuencia de la portadora, se encuentran los algoritmos que tienen una mejor aproximación al MCRB (Límite Modificado de Cramer-Rao, es un límite teórico al cual se espera que los recuperadores de frecuencia lleguen).

  Consideraciones de los algoritmos: Se conocen los símbolos de los datos de los cuales se recuperará la frecuencia; \( \hat{f} \) (estimado del desplazamiento en frecuencia), es un valor más pequeño que \( \frac{1}{T} \) Por así decirlo, son los algoritmos que hacen uso de toda la información disponible para poder obtener el mejor estimado de frecuencia.

  Algunos métodos son:
  - Método Kay.
  - Método Fitz.
  - Método Luise and Regiannini.
  - Método de Aproximación de la Estimación de la Máxima Similitud.

- **Sin Datos Auxiliares pero con Recuperador de Reloj Auxiliar.** De los recuperadores de frecuencia de la portadora, no se encuentran tan cercanos como los métodos involucrados en la clasificación anterior. Sin embargo, están más cerca del MCRB que las clasificaciones siguientes.

  Consideraciones de los algoritmos: Se conoce la información de la temporización, suponiendo una temporización ideal; condición de la tasa de Nyquist se sigue siendo un valor más pequeño que \( \frac{1}{T} \). Los métodos revisados son:
  - Algoritmo de Lazo Cerrado.
  - Algoritmo de Lazo Abierto, llamado "Retardo y Multiplicación".


• **Recuperador de Lazo Cerrado sin Información de la Temporización.** De los recuperadores de frecuencia de la portadora, son los más reales ya que no existe información alguna que pueda ayudar a los mismos algoritmos. Consideraciones: v puede obtener valores en el orden de la tasa del símbolo; no hay información de temporización.

• **Recuperador de Lazo Abierto sin Información de la Temporización.** Parte con las mismas características y condiciones que la clasificación pasada.

  o **Método de Retardo y Multiplicación.**

### 1.4.2 Algoritmos de Estimación de la Frecuencia con Datos Auxiliares

El funcionamiento de los algoritmos aquí tratados, depende de parámetros tales como:

- Cociente de Señal a Ruido (SNR)
- Longitud de la Observación $L_K$. Es la longitud del intervalo de observación en los periodos de los símbolos.
- Parámetro $N$ (en los estimadores de Fitz y L&R).

Todos los algoritmos que aquí se presentan, obtienen una estimación $\hat{f}$ del desplazamiento en frecuencia de la portadora [2].

#### 1.4.2.1 M étodo Kay [2].

La fórmula de estimación de Kay es la siguiente:

$$\hat{v} = \frac{1}{2\pi T} \sum_{k=1}^{L_K-1} \gamma(k) \arg\{z(k)z^*(k-1)\}$$  \hspace{1cm} (1)

Donde la función $\arg\{\ast\}$ es la fase, y $\{j(k)\}$ es una función suavizante dada por:

$$\gamma(k) = \frac{3}{2} \frac{L_0}{L_0-1} \left[ 1 - \left( \frac{2k-L_0}{L_0} \right)^2 \right] \hspace{1cm} k = 1, 2, ..., L_0 - 1$$  \hspace{1cm} (2)

Éste estimador es imparcial y busca el límite modificado de Cramer-Rao.

#### 1.4.2.2 M étodo Fitz [2].

Se llamará $R(m)$ como las autocorrelaciones de $z(k)$ obtenidas de los datos de $'z(k)'$. La fórmula de estimación de Fitz es la siguiente:

$$\hat{v} = \frac{1}{\pi \sqrt{(N+1)T}} \sum_{n=-N}^{N} \arg\{R(m)\}$$  \hspace{1cm} (3)
Donde la función \( \arg(*) \) es la fase de las correlaciones.

El estimador de Fitz se encuentra imparcial para el rango de

\[ \frac{L_0}{2} \leq N \leq \frac{1}{4NT} \]

y logra lo que el MCRB(v) además de que \( N \) es igual a \( \frac{L_0}{2} \). La exactitud de su estimación se degrada conforme \( N \) se decrementa pero, al mismo tiempo, la carga de cálculos es más ligera y el rango de estimación se hace más amplio. Así, hay una compensación entre el rango de estimación, por una parte, y una simplicidad en el cálculo y de exactitud por la otra.

### 1.4.2.3 Método de Luise & Regiannini [2].

A continuación se muestra el diagrama bloques de este estimador, que tiene a su entrada un filtro acoplado:

[Diagrama de bloques]

La fórmula de estimación de Luise & Regiannini es la siguiente:

\[
\hat{v} = \frac{1}{\pi (N + 1)T} \arg \left\{ \sum_{m=1}^{N} R(m) \right\}
\]

(4)

Para \( y \leq \frac{1}{NT} \), el estimador L&R utiliza una tabla para calcular los argumentos de la suma. Es imparcial en el rango y también logra el MCRB(v) en valores de \( N \) tan bajos como OdB y para \( N \approx L_0/2 \).

Como sucede con el estimator Fitz, la exactitud decrece lentamente conforme \( N \) decrece cuando el rango de estimación se hace más amplio.

### 1.4.2.4 Método de la Aproximación de la Estimación de la Máxima Similitud [2].

La aproximación deseada es:

\[
\Gamma(\hat{v}) = \sum_{k=1}^{L_0} z(k) e^{-j2\pi kT}
\]

(5)

De la búsqueda rugosa (o búsqueda con valores grandes) se puede encontrar su valor máximo mediante la Transformada Rápida de Fourier (FFT).
1.4.2.5 Comparaciones de Funcionamiento [2]

Características individuales:

- El método Kay fallará considerablemente a comparación de los otros algoritmos aquí mencionados si la Relación Señal a Ruido (SNR) decrece.
- Los métodos de Fitz y L&R son aproximadamente iguales para una TV dada. Ambos proveen una compensación entre la exactitud y la complejidad de la implementación.
- Los métodos de Fitz y L&R requieren de \( \frac{N(2L_0 - N - 1)}{2} \) multiplicaciones complejas (donde N es el número de elementos, y Lo es el paso de observación). Mientras que el Método de ML (Máxima Similitud) requiere de \( \left( \frac{1}{2} ML_0 \right) x \log_2 (ML_0) \) multiplicaciones complejas (donde M es llamado como el "factor de relleno de ceros" y que es propio del algoritmo de ML; Lo es el paso de observación). La anterior parte de forma general: \( M = 4 \) cuando \( L_0 < 64 \) y en caso contrario \( M = 2 \).

1.4.3 Algoritmo de Estimación Sin Datos Auxiliares pero con Recuperador de Reloj Auxiliar.

1.4.3.1 Algoritmo de Lazo Cerrado [2].

Aquí, las muestras del futuro acoplado son multiplicadas por \( c_i \) (que son las decisiones complejas-conjugadas del detector), las decisiones complejas conjugadas del detector, y se alimentan a un generador de error. El propósito del generador es el de dar una indicación de la diferencia entre \( v \) y su estimado correspondiente \( v(t) \), proporcionado por el Oscilador Controlado de Voltaje (VCO). La señal de error es filtrada (para atenuar el efecto del ruido) y usada para dirigir la frecuencia de VCO hacia \( v \). En particular, la frecuencia oscilante \( v(t) \) en los casos anteriores, iguala la diferencia entre la frecuencia oscilante y la frecuencia de corrida libre ("free running frequency") en el último. El desplazamiento \( v \) sigue bajo acción de la señal \( u(k) \) proporcionada por el filtro de lazo. Cuando \( v(t) \) es menor que \( v \), \( u(k) \) tiene componentes de DC positivas y el VCO es forzado a que aumente la velocidad. De forma similar, cuando \( v(t) \) es mayor que \( v \), VCO es forzado a ir más lento.

![Diagrama de bloques de la generación del error en frecuencia.](image)
1.5 RECUPERACIÓN DE LA FASE DE LA PORTADORA

Existe un gran número de recuperadores de fase de portadora, debido a la consecuencia de varios escenarios que han sido pensados, dependiendo del tipo de modulación y de la disponibilidad de información sea datos o de Temporización. Partiendo de lo anterior, podría verse de la siguiente forma:

(i)  Tipo de Modulación

(ii) Conocimiento Adicional
    a. Disponibilidad o no de la información de la temporización.
    b. Disponibilidad o no de la información de los símbolos. Cuando existen, deben provenir de un preámbulo conocido (esquemas de datos auxiliares) o de la salida del detector (esquema de decisión dirigida por datos).

(iii) Topología del Estimador
    a. Estimadores podrían bien ser de lazo cerrado o de lazo abierto.

Otra distinción proviene de la presencia de desplazamientos en frecuencia de la portadora. Sin embargo, estos esquemas tienden a ser realmente complejos, por simplicidad se asumirá que la recuperación de la portadora ha sido obtenida previamente. El hecho anterior no limita a que varios estimadores de fase de portadora no puedan enfrentar pequeños desplazamientos en frecuencia de la portadora. Sin embargo, los errores en frecuencia no son siempre moderados en una aplicación real. Cuando esto sucede, es necesario dotar al estimador de la fase de la portadora con información extra sobre la frecuencia de la portadora.
2. PLANTEAMIENTO DEL PROBLEMA

El problema nace debido a la necesidad del proyecto que se esta desarrollando en el CINVESTAV con respecto al Sistema de Comunicaciones Digitales, siendo más específicos en el receptor.

Se esta desarrollando un Sistema de Comunicaciones Digitales en el CINVESTAV Unidad Zacatenco, en el Departamento de Comunicaciones a cargo del Dr. Aldo Gustavo Orozco Lugo. La necesidad es el construir un receptor digital que en su etapa de demodulación este orientada a un esquema de modulación QAM, así como del tipo de detección que es coherente y con modo de transmisión en ráfaga.

Como se habla de un receptor de detección coherente, es necesario el obtener una fase, por lo que la Recuperación de Portadora para la estimación en frecuencia y fase es necesaria. Los diagramas a bloques que se han planteado para éste Sistema de Comunicaciones Digital en específico son los siguientes:

![Diagrama de bloques del transmisor](image1)

**Figura 2.1 Transmisor del DCS.**

El transmisor genera datos a razón de 500 símbolos por segundo. Los datos se modulan en PSK-4, para después ser acondicionados para ser lanzados al canal.

![Diagrama de bloques del receptor](image2)

**Figura 2.2 Receptor del DCS.**

Note que el bloque del Receptor queda abierto, lo anterior es debido a que el presente Trabajo Terminal se enfoca a la entrada del mismo, por ello no se dará más descripción de los módulos siguientes.

En el receptor se cuenta como primer elemento al acondicionador de señal de entrada, posteriormente se demodula la señal PSK-4. A la salida de la demodulación se tiene el primer algoritmo de recuperación de portadora, el cual obtiene un estimado en frecuencia (burdo) para después de-rotar a la misma señal. El Recuperador de Temporización estima y corrige el error de sincronización, posteriormente el segundo recuperador de portadora, obtiene un estimado en frecuencia (fino) en base a la sincronización obtenida anteriormente. Finalmente, el recuperador de fase de portadora corrige la fase de la misma. Por tanto, a continuación se muestran las características de los algoritmos mediante su descripción, teniendo como algoritmos seleccionados según el tipo de transmisión de datos en modo ráfaga a:
• "Retardo y Multiplicación sin Información"
• "Retardo y Multiplicación con Información de Temporización"
• "Estimación Adelantada con QAM"

Los dos primeros son algoritmos para obtener una estimación del desplazamiento en frecuencia de la portadora, el tercero es un amarrador de fase.

2.1 Algoritmo de "Retardo y Multiplicación sin información".

Es una técnica de recuperación de frecuencia de lazo abierto, sin información de temporización. La implementación digital del esquema Multiplicación-Retardo en la figura 1.5 será como se menciona a continuación.

![Esquema de Retardo y Multiplicación sin Información.](image)

Empecemos con el ancho de banda del \( LPF \) y observe que:

i) \( B_{LPF} \) debe ser enfocado a la señal \( s(t) \) (señal original) y no a la señal recibida con el ruido aditivo inmerso \( r(t) \). Así, \( B_{LPF} > F \) debe ser los suficientemente grande para que \( s(t) \), pase sin distorsión aún cuando el desplazamiento en frecuencia esta en su máximo, digamos \( \pm v_m \).

ii) El ancho de banda de la señal debe ser igual a \( \frac{1+a}{2T} \)donde \( a \) es el factor de rolloff. Así, para \( v_{max} \) en el orden de \( \frac{1}{2} \) de \( \frac{1}{T} \) ancho de banda LPF aproximadamente a

Como \( x(t) \) esta dentro de la banda limitada \( \pm \frac{2}{T} \) igue que \( z(t) = x(t)x^*(t-\Delta T) \) es banda limitada y esta dentro de \( \pm \frac{4}{T} \). De la fórmula propuesta para el caso analógico:

\[
\hat{v} = \frac{1}{2\pi \Delta T} \arg \left\{ \int_{0}^{T} z(t)dt \right\}
\]

(donde \( P \) es el estimado de frecuencia y la función \( \arg^* \) es la fase), se considera que \( TQ \) sea muchísimo más grande que \( T \), la integral puede ser calculada de las muestras de \( z(t) \) tomando una tasa de \( R = \frac{4}{T} \).
Formalmente

\[
\int_0^{t_0} z(t) dt = \frac{T}{4} \sum_{i=0}^{4^{L_0}-1} x \left( \frac{kT}{4} + t_o \right) x^* \left( \frac{kT}{4} + t_o - \Delta T \right)
\]  

(2)

Donde \( L_0 = \frac{T_0}{T} \) (Longitud de Observación) y \( t_o \) es una fase de muestreo arbitraria. \( R \) no necesita ser realmente \( \frac{4}{T} \). Pequeñas desviaciones de este valor son equivalentes a un barrido periódico \( t_o \) entre 0 y aproximadamente constante sobre \( T_o \) segundos, entonces la fórmula anterior se mantiene válida.

Ajustando \( t_o = 0 \) por simplicidad y substituyendo la ecuación 11 en la ecuación 12 se genera el algoritmo de estimación deseado.

\[
\hat{v} = \frac{1}{2\pi\Delta T} \arg \left\{ \sum_{i=0}^{4^{L_0}-1} x \left( \frac{kT}{4} \right) x^* \left( \frac{kT}{4} - \Delta T \right) \right\}
\]  

(3)

En general la suma en (3) implica 8\( L_i \) muestras de \( x(t) \). Este número podría ser dividido en dos, en \( \Delta T = \frac{T}{4} \) En particular, escogiendo \( \Delta T = \frac{T}{4} \), resulta en:

\[
\hat{v} = \frac{2}{\pi T} \arg \left\{ \sum_{i=0}^{4^{L_0}-1} x \left( \frac{kT}{4} \right) x^* \left( \frac{(k-1)T}{4} \right) \right\}
\]  

(4)

La figura 1.6 muestra un diagrama a bloques del algoritmo (3). Que será la implementación digital del Método de Retardo y Multiplicación.

Aquí, \( x(k) \) significa \( x(kT/4) \) y \( z^{-1} \) representa un retardo de \( \frac{T}{4} \), Aclarando que, \( x(k) \) se usa de forma indistinta \( \cos(t_i) \).

Cabe mencionar, que lo que se encuentra es un estimado de la frecuencia. El "Método de Retardo y Multiplicación" es un método de lazo abierto, el cuál es rápido para el modo de transmisión en ráfaga.

Figura 2.4 Diagrama a bloques para el estimador digital.
El análisis del funcionamiento de este estimador es complejo y es perseguido en [4]. La figura ilustra los resultados de la simulación para la varianza de la estimación con señalización PSK-4. La curva ha sido dibujada para $v = 0$ pero los mismos resultados son obtenidos para cualquier $v$ sobre el intervalo de $\pm \frac{1}{2} T$.

Donde $L_0$ es el número de muestras tomadas por el algoritmo. C.M. es el número de “Corridas Montecarlo” realizadas [Apéndice A]. Alpha, es el exceso de ancho de banda del sistema.

Si se compara con la figura del estimator del desplazamiento en frecuencia siguiente, se observa como claramente el presente algoritmo es superado. No hay que olvidar que el algoritmo de “Retardo y Multiplicación con Información de Temporización” trabaja a distinto nivel que el descrito en esta sub-sección.
2.2 Algoritmo de "Retardo y Multiplicación Con Información De Temporización" [2].

Esquemas de lazo abierto son más atractivos para modos de transmisión de ráfaga, debido a sus cortos tiempos de estimación. Para una revisión más profunda de este algoritmo diríjase a [5]. Para este algoritmo debe tomarse en consideración que v debe ser una pequeña fracción de la tasa del símbolo. Ahora, a la salida del filtro acoplado es aproximadamente:

$$y(k) = c_k e^{j[2\pi(i_f+i)+\varphi]} + n(k)$$  \hspace{1cm} (5)

Observe que en una constelación QPSK \( \{e^{j(m \pi/2)}, m = 0, 1, 2, 3\} \). Se verifica que \( c_k^4 = 1 \). Así, elevando a la cuarta potencia obtenemos

$$y^4(k) = c_k e^{j[8\pi(i_f+i)+4\varphi]} + n'(k)$$  \hspace{1cm} (6)

Donde \( n'(k) \) es un término del ruido resultado de las interacciones RuidoXRuido. Se observa que la modulación ha sido removida de \( y^4(k) \) por \( [y^*(k-1)]^4 \) obtenemos:

$$[y(k)y^*(k-1)]^4 = e^{j8\pi\varphi} + n''(k)$$  \hspace{1cm} (7)

Donde nuevamente \( n''(k) \) proviene de los productos de SeñalXRuido y RuidoXRuido. La ecuación anterior indica que \( [y(k)y^*(k-1)]^4 \) es un estimado de \( e^{j8\pi\varphi} \). La exactitud de la estimación puede ser mejorada mediante la eliminación de las componentes del ruido como a continuación se muestra:

$$\frac{1}{L_0-1} \sum_{k=1}^{L_0-1} [y(k)y^*(k-1)]^4 = e^{j8\pi\varphi} + \frac{1}{L_0-1} \sum_{k=1}^{L_0-1} n''(k)$$  \hspace{1cm} (8)

Finalmente, asumiendo que el último término en (4.17) es pequeño en amplitud comparado con la unidad y tomando el argumento de ambos lados, tenemos:

$$\hat{v} = \frac{1}{8\pi f} \arg \left\{ \sum_{k=1}^{L_0-1} [y(k)y^*(k-1)]^4 \right\}$$  \hspace{1cm} (9)

A continuación se muestra el diagrama bloques de este estimador:

![Diagrama de bloques](image)

Figura 2.6 Diagrama a bloques de un estimator de lazo abierto.
El \( \arg\{\cdot\} \) del algoritmo toma valores en el rango de \( \pm \pi \) y el estimado del desplazamiento varía entre \( \pm \frac{1}{8T} \). Es por lo anterior que el diagrama a bloques no tiene un bloque que multiplique por la constante que se encuentra en el algoritmo.

Del diagrama a bloques se observa como la señal de entrada pasa a un multiplicador, pero que la misma toma una ramificación para ser retardada una unidad y para después ser conjugada. Una vez que se tienen las funciones a la entrada del multiplicador, la operación se realiza. Después se eleva a la cuarta potencia la multiplicación antes hecha para que finalmente la función \( \arg\{\cdot\} \) sea la fase de la suma. Para así obtener el estimado del desplazamiento de frecuencia.

La siguiente figura muestra la estimación de la varianza en función de \( E_b/\text{No} \) cuando un verdadero desplazamiento en frecuencia ha sido de 0 o el 10% de la tasa del símbolo. Debe hacerse notar que esta estimación no está muy alejada del MCRB, que su gráfica está situada para 10dB de \( E_b/\text{No} \) en 4x10^{-7} en la escala logarítmica, así como en 10x10^{-8} para una \( E_b/\text{No} \) de 30dB.

![Figura 2.7 Varianza de la Frecuencia Normalizada en función de la \( E_b/\text{No} \).](image)

Se debe tener mucha consideración con respecto a la Varianza de la Frecuencia Normalizada del algoritmo "Retardo y Multiplicación" con información de temporización con respecto a la de su acrónimo pero sin información alguna.
2.3 Estimación Adelantada Con QAM (Recuperación de Fase) [2]

Este algoritmo adopta una aproximación heurística. A la salida de un filtro acoplado, se tiene lo siguiente:

\[ x(k) = c_x e^{j\theta} + n(k) \]  

(10)

Con QAM los símbolos tienen la siguiente forma \( c_k = a_k + jb_k \), donde \( a_k \) y \( b_k \) tienen una media cero y son variables aleatorias independientes con momentos de segundo y cuarto orden. El valor esperado puede encontrarse como:

\[ E[a_k^2] = E[b_k^2] = C_2 \]  

El valor esperado puede encontrarse como:

Como la ecuación \( C_4 - 3C_2^2 \quad E[x^4(k)] = 2(C_4 - 3C_2^2)e^{j4\theta} \)  

(11)

tiende a ser positiva, se dice que la ecuación anterior provee información de la fase de la portadora. Un hecho es:

\[ \theta = \frac{1}{4} \text{arg}\{E[x^4(k)]\} \]  

(12)

En la práctica, la operación esperada puede ser aproximada por el promedio de muestra de \( x^4(k) \), así la ecuación anterior, se transforma en el estimador:

\[ \theta = \frac{1}{4} \text{arg}\{\sum_{k=0}^{N-1} x^4(k)\} \]  

(13)

Este algoritmo también cumple para el caso especial QAM llamado QPSK, además, su desempeño logra lo que el MCRB para una relación señal a ruido muy grande. Con una modulación QAM, de hecho, la exactitud de su estimación decrece más y más con respecto al MCRB conforme el número de puntos en la constelación crece[6].

El algoritmo estimador de fase es conocido también como "Estimador de fase de Potencia M", lo anterior es, porque no solo aplica para este caso en particular, sino que si se desea estimar la fase de una señal de un número mayor de puntos en la constelación, el factor que se modifica en este caso es el número cuatro que está elevando a los símbolos a la entrada, así como el factor que divide a la función argumento de la suma de los N símbolos.

El presente algoritmo, cabe mencionar, presenta cierta ambigüedad ya que estima la fase para cuando el desplazamiento en fase no es mayor a 45 grados, de lo contrario, no se sabe con detalle donde se encuentran los valores correctos a menos que, en la etapa siguiente a la presente existiese un detector por secuencia de entrenamiento, o alguna otra técnica para poder determinar de forma correcta el símbolo; aclarando que el tema de detección de símbolo como tal, no se trata en el presente escrito.
A continuación se muestra el diagrama a bloques de éste estimador:

![Diagrama a bloques del Recuperador de Fase](image)

Figura 2.8 Diagrama a bloques del Recuperador de Fase

La siguiente figura muestra el desempeño del estimador en fase, la varianza del error en fase con respecto a la relación señal a ruido. Cabe mencionar que el presente algoritmo alcanza lo que el Límite Modificado de Cramer Rao logra en 17.5dB

![Gráfica de desempeño del estimador de fase](image)

Figura 2.9 Desempeño del estimador de Fase

### 2.4 SOLUCIÓN PROPUESTA

El seguimiento de la solución propuesta debe ser la figura 2.2, ya que es el DCS propuesto por el proyecto mismo. Ahora bien, el objetivo principal es el que los algoritmos de recuperación de Frecuencia y Fase de la Portadora sean implementados en un DSP. Se eligió de la compañía Texas Instruments el DSP de la familia 2000, el TMS320F2812 debido a sus cualidades físicas que lo hacen más sencillo de implementar en una tarjeta independiente. Para su programación, Texas Instruments, comercializa junto con la tarjeta de desarrollo un entorno de desarrollo llamado Code Composer Studio (CCS), en el cual se puede llevar a cabo desde la programación del DSP hasta la verificación de datos en tiempo real, ya sean en tablas o gráficas.
3. DESARROLLO DE LA PROPUESTA

Durante el presente capítulo, se mostrarán todos los elementos que son necesarios para desarrollar la propuesta.

El presente capítulo se subdivide en:

1) Diagramas de flujo. 
Son necesarios para la programación de los algoritmos, tanto en simulación como para la programación final en el DSP.

2) Simulaciones de los distintos algoritmos a implementar. 
Son necesarios para darse una idea del desempeño de los mismos algoritmos, así como de las características que toman en cuenta para tener un buen desempeño, para lo cual, se simularon los siguientes algoritmos:
   a) "Retardo y Multiplicación sin Información"
   b) "Retardo y Multiplicación con Información de Temporización"
   c) "Estimación Adelantada con QAM"

3) Introducción a los DSP. 
Aquí se trata a grandes rasgos qué y porqué un DSP.

4) DSK o DSP Starter Kit
Se explica brevemente lo que es la tarjeta de desarrollo de Texas Instruments fabricada por Spectrum Digital.

5) Code Composer Studio
Se da una explicación breve de lo que es, así como de las grandes ventajas que tiene este entorno de desarrollo.

6) DSPTMS320F2812
Se da una explicación teórica de la arquitectura tomada en cuenta para la programación de los algoritmos en el presente Trabajo Terminal. También se hace mención de los elementos que se tomaron en cuenta durante el desarrollo de la implementación de los algoritmos en el DSP.

7) Interfaz serial DSP-MATLAB.
Se muestran los diagramas de flujo de las funciones encargadas de comunicar al puerto serial tanto de MATLAB como del DSP, así como del protocolo de transmisión serial empleado.

8) Diseño y Fabricación de la Tarjeta Independiente.
Los diagramas esquemáticos así como los del circuito impreso se encuentran en esta sección.
En los siguientes apartados, se hará una descripción más profunda sobre los temas anteriores.
3.1 Diagramas de Flujo

En este apartado se muestran los diagramas de flujo utilizados para simulación de los algoritmos de Recuperación de Portadora, así como del algoritmo de Recuperación de Fase. Las simulaciones se llevaron a cabo en el programa MATLAB-7.

Figura 3.1 Diagrama de Flujo del algoritmo “Retardo-Multiplicación sin Información”
Figura 3.2 Diagrama de Flujo del Algoritmo "Retardo y Multiplicación con l. de T."

REPUPERADOR DE PORTADORA DE UN RECEPTOR DIGITAL
CON MODULACIÓN QAM IMPLEMENTADO EN UN DSP
Los diagramas de flujo anteriores son los utilizados para las simulaciones que a continuación se mostrarán, así como para el código escrito para el DSP.
3.2 Simulaciones de los distintos Algoritmos a Implementar

3.2.1 Simulaciones del algoritmo "Retardo y Multiplicación sin información"

Las simulaciones siguientes muestran el compromiso que tiene el algoritmo con parámetros como: la longitud de observación (elementos de procesamiento), el desplazamiento en frecuencia y el factor de roll-off para un esquema de modulación PSK-4. El pulso \( g(t) \) corresponde a un coseno alzado.

Para poder lograr las curvas aquí mostradas es necesario realizar "Corridas de Montecarlo", véase Apéndice A.

Para valores de \( v \) que están dentro del intervalo \( \pm \frac{1}{T} \) las gráficas deben ser las mismas, es decir, que para desplazamientos en frecuencia \( f_o \) que estén dentro de este intervalo no debe existir diferencia de operación entre todos ellos en el algoritmo [2]. Para lo anterior, se propone una \( T = 1 \) para comparar con distintos factores del mismo intervalo.

De esta primera figura se puede observar que después de 150 Corridas Montecarlo, la curva no ha llegado a suavizarse tanto como se esperaba. Se ve como el algoritmo a distintos valores de desplazamiento en frecuencia (\( f_o \) en la gráfica) tiene el mismo comportamiento, confirmando lo que en la literatura se hace mención.
Figura 3.5 Varianza del Error del Estimador "Retardo-Multiplicación sin información con 150 símbolos y con exceso de ancho de banda de 0.25"

Figura 3.6 Varianza del Error del Estimador "Retardo-Multiplicación sin información con 200 símbolos y con exceso de ancho de banda de 0.25"

Figura 3.7 Varianza del Error del Estimador "Retardo-Multiplicación sin información con 300 símbolos y con exceso de ancho de banda de 0.25"
Para las figuras ya mostradas, se tiene un factor de roll-off de 0.25, se observa que no hay gran diferencia de comportamiento de las curvas debido al factor de exceso de ancho de banda. Lo que es notorio es la suavización de las curvas conforme el intervalo de observación crece.

Aparentemente para pequeños valores en la relación señal a ruido (SNR) la estimación es buena, ya que para distintos desplazamientos en frecuencia el comportamiento del algoritmo debe ser el mismo, sin embargo, existe una separación notoria cuando la SNR es de valor 10.

Nótese los niveles mínimos alcanzado por el algoritmo "Retardo y Multiplicación con Información de Temporización".

Se mostrarán las siguientes gráficas donde el factor de roll-off cambia para observar el efecto de los mínimos de las curvas conforme el SNR crece.
Figura 3.10 Varianza del Error del Estimador “Retardo-Multiplicación sin información con 200 símbolos y con exceso de ancho de banda de 0.5”

Figura 3.11 Varianza del Error del Estimador “Retardo-Multiplicación sin información con 300 símbolos y con exceso de ancho de banda de 0.5”

Figura 3.12 Varianza del Error del Estimador “Retardo-Multiplicación sin información con 100 símbolos y con exceso de ancho de banda de 0.75”
Figura 3.13 Varianza del Error del Estimador “Retardo-Multiplicación sin información con 150 símbolos y con exceso de ancho de banda de 0.75”

Figura 3.14 Varianza del Error del Estimador “Retardo-Multiplicación sin información con 200 símbolos y con exceso de ancho de banda de 0.75”

Figura 3.15 Varianza del Error del Estimador “Retardo-Multiplicación sin información con 300 símbolos y con exceso de ancho de banda de 0.75” 300
El desempeño del algoritmo "Retardo y Multiplicación sin Información" mejora conforme el exceso de ancho de banda es mayor. La varianza del error normalizado disminuye notablemente de 5dB a 15dB de la relación señal a ruido.

La siguiente gráfica muestra la salida del algoritmo de "Retardo y Multiplicación sin Información". Cabe recordar que el algoritmo trabaja con muestras de — del símbolo, por ende si la entrada de fue de 100 símbolos el estimado de frecuencia lo realiza con 400 muestras.

A la salida del mismo, se observan las cuatrocientas muestras que han sido ya corregidas de forma burda por el algoritmo "Retardo y Multiplicación sin Información", y se menciona burda, ya que este estimador de frecuencia llega a un estimado no muy preciso, claro esta, debido a la estimación sin ayuda. Un ejemplo de lo anterior, sería el siguiente:
Si se ha obtenido un estimado en desplazamiento en frecuencia de 0.01Hz y se cuentan con cien símbolos de entrada, entonces existen 400 muestras; como el algoritmo hace la corrección sobre las muestras, es entonces cuando se entiende que por cada cien muestras, ya se habrá dado una vuelta entera.

La gráfica anterior es el resultado de la simulación del algoritmo en MATLAB, con un esquema de modulación PSK-4. La gráfica anterior nos muestra las muestras de salida del algoritmo de "Retardo y Multiplicación sin Información".
3.2.2 Simulaciones del algoritmo "Retardo y Multiplicación con información de Temporización"

Las siguientes simulaciones representan al algoritmo de "Retardo y Multiplicación con Información de Temporización". Debe recalzarse que existe una gran diferencia entre los algoritmos de Recuperación de Portadora aquí vistos, y es con el tipo de datos que trata el algoritmo; El algoritmo que no hace uso de ningún tipo de información hace su estimación de desplazamiento de frecuencia en base a muestras tomadas del muestreador, las cuales deben ser de un cuarto del periodo del símbolo. Por el otro caso, el algoritmo que hace uso de la información de la temporización, su procesamiento lo realiza sobre los datos interpolados que la etapa anterior al sistema (Recuperador de Reloj) le entrega a su salida.

Ambos son de gran importancia sobre un sistema de detección coherente, ya que el primer algoritmo (sin información) hace un estimado burdo del desplazamiento de la portadora y el segundo (con información de la temporización) hace un estimado fino del desplazamiento de la portadora. Lo último se observa en las siguientes simulaciones.

Como se observa en la figura anterior, ahora se entiende el porque el algoritmo de recuperación de portadora sin información hace un estimado burdo, lo anterior se refiere a que no puede realizar una estimación tan fina como lo hace el algoritmo de recuperación de portadora con información del temporizador. El algoritmo "Retardo y Multiplicación sin información" es aquel de color negro y el algoritmo "Retardo y Multiplicación con información de Temporización" son las demás gráficas.
Figura 3.18 Varianza del Error del Estimador "Retardo-Multiplicación con I. de T. con 200 símbolos y con un exceso de ancho de banda de 0.25"

Figura 3.19 Varianza del Error del Estimador "Retardo-Multiplicación con I. de T. con 300 símbolos y con un exceso de ancho de banda de 0.25"

Figura 3.20 Varianza del Error del Estimador "Retardo-Multiplicación con I. de T. con 100 símbolos y con un exceso de ancho de banda de 0.5"
Figura 3.21 Varianza del Error del Estimador "Retardo-Multiplicación con I. de T. con 200 símbolos y con un exceso de ancho de banda de 0.5"

Figura 3.22 Varianza del Error del Estimador "Retardo-Multiplicación con I. de T. con 300 símbolos y con un exceso de ancho de banda de 0.5"

Figura 3.23 Varianza del Error del Estimador "Retardo-Multiplicación con I. de T. con 100 símbolos y con un exceso de ancho de banda de 0.75"
Figura 3.24 Varianza del Error del Estimador “Retardo-Multiplicación con I. de T. con 200 símbolos y con un exceso de ancho de banda de 0.75”

Figura 3.25 Varianza del Error del Estimador “Retardo-Multiplicación con I. de T. con 300 símbolos y con un exceso de ancho de banda de 0.75”

Figura 3.26 Varianza del Error del Estimador “Retardo-Multiplicación con I. de T. con 100 símbolos y con un exceso de ancho de banda de 1.0”
Se observa de las gráficas que el algoritmo tiene un mejor desempeño conforme el número de elementos a procesar es mayor, y no es raro, ya que la estimación del desplazamiento de la portadora para este algoritmo tiene que ver con los mismos elementos que se elevan a la cuarta, así mejorando su estimación.

Cabe mencionar que el algoritmo “Retardo y Multiplicación con información de Temporización” se acerca en comportamiento y niveles al MCRB, que siempre es buscado. Por lo que se ve el nivel de desempeño del algoritmo en cuestión, conforme a la estimación de frecuencia.

Ahora bien, la siguiente gráfica muestra los símbolos que se obtienen del estimador de frecuencia "Retardo y Multiplicación con Información de Temporización" a la salida.

El estimador de frecuencia del algoritmo "Retardo y Multiplicación con Información de Temporización" obtiene valores tan pequeños, que realmente no tiene problema alguno para poder corregir en frecuencia a la señal recibida, lo anterior se observa con la siguiente
gráfica, donde se observan los datos perfectamente recuperados, para un tipo de modulación PSK-4. Cabe hacer mención, que el transmisor envía la señal PSK con fase inicial de 45 grados, es entonces cuando se observa en la gráfica que existe un defasamiento pequeño, menor a 45 grados, que será importante para el siguiente algoritmo.

La diferencia de la estimación de frecuencia del algoritmo que hace uso de la información del reloj con respecto al que no utiliza información alguna, es grande, y se ha observado en las gráficas anteriores, donde el desempeño del algoritmo "Retardo y Multiplicación con Información de Temporización" es en definitiva mejor. Sin embargo, éste algoritmo no podría por si solo obtener el desempeño anterior, ni aún así con ayuda de un recuperador de temporización (a menos de que éste último recupere el reloj aún en presencia de un gran desplazamiento en frecuencia) si existe un gran desplazamiento en frecuencia; considerando que en la realidad, los desplazamientos en frecuencia son muy grandes por diversas circunstancias.

La figura obtenida a la salida del algoritmo de "Retardo y Multiplicación con Información de Temporización" al igual que la mostrada en la sección anterior fue obtenida mediante la simulación del algoritmo en MATLAB, para una modulación PSK-4.
3.2.3 Simulaciones del algoritmo "Estimación Adelantada con QAM"

Las siguientes simulaciones nos ayudan a comprender el desempeño del algoritmo de Estimación Adelantada con QAM dentro del intervalo de fase de cero a 45 grados, en el cual no existe diferencia alguna [2], sin embargo, pruebas realizadas con defasamientos más grandes no generan las siguientes gráficas, creando una ambigüedad en la respuesta del algoritmo a la salida de la señal PSK-4 utilizada para el ejemplo, es decir, cuando existe un defasamiento mayor de 45 grados el algoritmo rota la constelación al siguiente eje (en caso de que el algoritmo de Estimación adelantada con QAM, haya puesto los puntos de la constelación en los ejes coordenados), y que al parecer recuperase la fase de la portadora, pero no es así.

A continuación mostramos las gráficas que muestran el desempeño del algoritmo a distintos defasamientos dentro del intervalo de 0 a 45 grados, cabe mencionar que en [2] se hace mención que el algoritmo toma en cuenta una recuperación de frecuencia de portadora, así como de una recuperación de reloj como utópicos.

![Gráfica de simulaciones del algoritmo "Estimación Adelantada con QAM"](image-url)

Figura 3.30 Varianza del Error en fase del Estimador de Fase para 100 símbolos.

En la gráfica anterior se analiza el desempeño del algoritmo para distintos cambios en fase (dF) que se encuentran dentro del rango permitido.

Nótese el desempeño del algoritmo para N símbolos tomados. Y el cómo se cumple la condición del rango de 0 a 45 grados.
Figura 3.31 Varianza del Error en fase del Estimador de Fase para 200 símbolos.

Figura 3.32 Varianza del Error en fase del Estimador de Fase para 300 símbolos.

Figura 3.33 Varianza del Error en fase del Estimador de Fase para 400 símbolos.
Como conclusión de las gráficas anteriores, se puede observar que mientras mayor sea el número de símbolos tomados para estimar la fase, mejor desempeño tendrá el algoritmo. Ya que la varianza del error en fase se ve disminuido contra la relación señal a ruido. Así mismo, no existe diferencia alguna si los defasamientos están dentro del rango de 0 a 45 grados, para que no exista una ambigüedad de fase; lo anterior quiere decir, que en dado caso que exista un defasamiento mayor, el algoritmo por sí mismo mandará a los puntos de las constelaciones a la referencia más cercana. La recuperación de fase puede ser vista en la siguiente figura.
Las gráfica anterior fue obtenida de la simulación del algoritmo "Estimación Adelantada con QAM" en MATLAB con un esquema de modulación PSK-4.

Como se mencionó anteriormente, puede llegar a existir cierta ambigüedad con el presente algoritmo, para poder evadir el problema anterior la literatura propone que durante la transmisión se envíe una secuencia de entrenamiento para que después de la etapa de recuperación de fase se pudiesen obtener los datos en fase correcta.

3.2.4 MATLAB

MATLAB es un lenguaje computacional y técnico de alto nivel, para desarrollo de algoritmos de una forma interactiva y que permite la visualización, análisis y cálculo de datos.

MATLAB se utiliza por las siguientes cuestiones:

1) MATLAB se enfoca en utilizar a la computadora como una herramienta para el análisis científico y de ingeniería.
2) Un lenguaje de alto nivel es el más apropiado para los cálculos científicos. MATLAB provee un gran número de funciones para operaciones complejas que son comunes. Así como de rutinas para la visualización de resultados, ya sea desde una constante hasta gráficas que serían muy complejas de realizar en otro lenguaje.
3) MATLAB tiende a ser el lenguaje más utilizado para los cálculos científicos.
4) Como es un lenguaje de alto nivel, el programador no tiene porqué preocuparse por el tamaño de los arreglos, operaciones, etc.
3.3 Introducción a los DSP’s

3.3.1 ¿Qué es un DSP?

Un Procesador Digital de Señales (DSP) es un tipo de microprocesador, uno que es rápido y poderoso. Un DSP es único por su proceso de datos en tiempo real. Esta capacidad hace a un DSP perfecto para aplicaciones que no pueden tolerar retardos. Como gran ejemplo, se tienen a los celulares de hoy en día, con los cuales ambas personas se pueden estar comunicando sin problema alguno, bueno pues, los celulares de hoy en día utilizan un DSP como procesador central. Los DSP’s que se encuentran dentro de los celulares, procesan sonidos tan rápidamente como los escuchas, así como tan rápidamente uno habla, es decir, en tiempo real [7].

3.3.2 ¿Por qué un DSP?

A continuación se mencionan algunas ventajas de la utilización de los DSP, sobre otros microprocesadores:

- Operación de Multiplicación-Acumulación en un ciclo máquina: Se debe a que la arquitectura del DSP permite realizar la multiplicación-acumulación en registros especializados y por tanto en un ciclo máquina.
- Funcionamiento en tiempo real, simulación y emulación: Las herramientas del DSP permiten simular y emular el programa cargado en el mismo, así como del funcionamiento en tiempo real, que se podrá verificar con dispositivos de medición tal y como lo es el osciloscopio, entre otros.
- Flexibilidad: Un DSP es flexible, ya que siendo un elemento programable es posible el programar el dispositivo y en caso de errores o mejoras a implementar, el DSP se puede programar nuevamente.
- Confiabilidad: Siendo el DSP un dispositivo digital, la confiabilidad es alta referida al procesamiento digital de señales.
- Escalabilidad: A diferencia de los dispositivos analógicos, un DSP siempre podrá ser escalable, quizás se implemente una pequeña etapa analógica para el DSP, pero que no tiene comparación con la realización de una tarjeta analógica completamente nueva para resolver el problema en cuestión.
- Costo reducido de los sistemas: Por lo mismo referido a la escalabilidad, un DSP trae consigo menores costos, ya que un DSP podría bien ser utilizado para resolver varios problemas.
3.4 DSK o DSP Starter Kit
La tarjeta de desarrollo "F2812 eZdsp" es una tarjeta, en la cual el DSP TMS320F2812 esta montado permitiendo la programación y simulación del mismo. Contiene todas las características físicas necesarias para el funcionamiento del DSP, así como de la interfaz que utiliza para comunicarse con la computadora y así con Code Composer Studio.

![Figura 3.37 DSP Starter Kit TMS320F2812 eZdsp.](image)

3.5 Code Composer Studio [7]
Diseñado por Texas Instruments (TI), Code Composer Studio es un entorno de desarrollo que integra de forma muy fina las capacidades de los siguientes componentes con una arquitectura abierta y por ende escalable.

1) Entorno de desarrollo integrado (Code Composer) con editor, compilador, administrador de proyecto (proyect manager), puntos de prueba y muchos otros.
2) Un compilador en lenguaje C, un optimizador de ensamblaje y de enlace (herramientas de generación de código).
3) Contiene un simulador de un conjunto de instrucciones.
4) Es un software fundamentado en tiempo-real.
5) Permite el intercambio de datos en tiempo-real entre el objeto de estudio (en este caso sería el Kit de Desarrollo del DSP) y la computadora.
6) Permite visualizar los datos así como de analizarlos en tiempo-real.
7) Cuenta con un módulo de monitoreo en tiempo-real para verificar el estado en el que se encuentra el DSP.
8) Su interfaz con el usuario es muy agradable.

Code Composer Studio integra todas las herramientas necesarias para mantener al Kit de desarrollo del DSP y a la computadora en un entorno unido. También, simplifica la configuración del sistema del DSP así como del diseño de la aplicación para ayudar a que los diseñadores empiecen a trabajar de forma más rápida a como anteriormente lo hacían.

![Figura 3.38 Presentación de Code Composer Studio](image)
3.6DSPTMS320F2812

El DSP TMS320F2812, es un DSP que pertenece a la familia 2000, que es de gran integración y que ofrece soluciones de gran desempeño para aplicaciones de control muy demandantes. Es claro que este DSP no es para aplicaciones de comunicaciones, sin embargo, ofrece ventajas claras en hardware sobre la familia 5000, que sus aplicaciones son para las comunicaciones móviles; así como de la 6000, que sus aplicaciones son para las comunicaciones terrenas.

La letra "F" indica que es un DSP de la familia de memorias FLASH. La memoria FLASH, nos permite programar el DSP en varias ocasiones, en vez de la ROM (letra "C", en lugar de la "F") que se programa por única ocasión.

La gran desventaja que tiene el DSP TMS320F2812, es con respecto a las operaciones que realiza, ya que las operaciones que se realizan son de punto fijo, para ello se tendrán consideraciones al momento de la programación del dispositivo. La programación simula las operaciones de punto flotante que se necesitan.

A continuación, se presenta una tabla que muestra las características principales del TMS320F2812:

<table>
<thead>
<tr>
<th>Característica</th>
<th>Valor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frecuencia de Operación Máxima (MHz)</td>
<td>150</td>
</tr>
<tr>
<td>RAM (Word)</td>
<td>18K</td>
</tr>
<tr>
<td>FLASH (Words)</td>
<td>128K</td>
</tr>
<tr>
<td>OTP ROM (Words)</td>
<td>1k</td>
</tr>
<tr>
<td>Cargador de Inicialización Disponible</td>
<td>SI</td>
</tr>
<tr>
<td>Interfase de Memoria Externa</td>
<td>SI</td>
</tr>
<tr>
<td>Canales PWM</td>
<td>16</td>
</tr>
<tr>
<td>Temporizadores</td>
<td>3</td>
</tr>
<tr>
<td>Total de Puertos Serials</td>
<td>CAN, McBSP, SPI, 2 SCIs</td>
</tr>
<tr>
<td>Alimentación Base (Volts)</td>
<td>1.9</td>
</tr>
<tr>
<td>Voltaje de E/S (Volts)</td>
<td>3.3</td>
</tr>
<tr>
<td>Descripción</td>
<td>Controlador de Señales Digitales con FLASH de 32 bits</td>
</tr>
<tr>
<td>Número de Canales de 12-bit A/D</td>
<td>16</td>
</tr>
</tbody>
</table>

Tabla 3.1 Características Generales del TMS320F2812.
El TMS320F2812 en arquitectura, es un DSP muy completo, ya que cuenta internamente con todos los elementos necesarios, desde el corazón del DSP que es el CPU, así como de las extremidades que son los puertos de entrada y salida, así como de las memorias, entre muchos otros componentes. La gran ventaja en arquitectura que tiene este DSP, es el de contar con Memoria Flash en el Chip, que es de 128K, así como de la interfaz que tiene para conectar memorias externas en dado caso que se necesite; cuenta con tres Temporizadores de CPU de 32 bits, un convertidor Analógico Digital de 12 bits, entre muchos otros módulos. Para mayor detalle observe la figura anterior.

El DSP TMS320F2812 presenta una arquitectura tipo Harvard, donde la memoria de datos así como la de programa están separadas.
3.6.2 Mapa de Memoria [8]

Memoria integrada en el chip:

<table>
<thead>
<tr>
<th>Espacio de datos</th>
<th>Espacio de programa</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vector M0 - RAM (32 x 32) (Habilitado si VMAP=0)</td>
<td></td>
</tr>
<tr>
<td>M0 SARAM (1K x 16)</td>
<td></td>
</tr>
<tr>
<td>M1 SARAM (1K x 16)</td>
<td></td>
</tr>
<tr>
<td>Periféricos del grupo 0 (2K x 16)</td>
<td></td>
</tr>
<tr>
<td>Vector PIE - RAM (256 x 16) (Habilitado si VMAP=1, ENPIE=1)</td>
<td></td>
</tr>
<tr>
<td>Reservado</td>
<td></td>
</tr>
<tr>
<td>Reservado</td>
<td></td>
</tr>
<tr>
<td>Periféricos del grupo 1 (4K x 16, Bloque Protegido)</td>
<td></td>
</tr>
<tr>
<td>Periféricos del grupo 2 (4K x 16, Bloque Protegido)</td>
<td></td>
</tr>
<tr>
<td>L0 SARAM (4K x 16, Bloque protegido)</td>
<td></td>
</tr>
<tr>
<td>L1 SARAM (4K x 16, Bloque protegido)</td>
<td></td>
</tr>
<tr>
<td>Reservado</td>
<td></td>
</tr>
<tr>
<td>OTP (o ROM) (1K x 16, Bloque protegido)</td>
<td></td>
</tr>
<tr>
<td>Reservado (1K)</td>
<td></td>
</tr>
<tr>
<td>Flash (o ROM) (128K x 16, Bloque protegido)</td>
<td></td>
</tr>
<tr>
<td>Palabra clave de 128 bit</td>
<td></td>
</tr>
<tr>
<td>H0 SARAM (8K x 16)</td>
<td></td>
</tr>
<tr>
<td>Reservado</td>
<td></td>
</tr>
<tr>
<td>Boot ROM (4K x 16) (Habilitado si MP/MC=0)</td>
<td></td>
</tr>
<tr>
<td>Vector BROM - ROM (32 x 32) (Habilitado si VMAP=1, MP/MC=0, ENPIE=0)</td>
<td></td>
</tr>
</tbody>
</table>

**Notas:**
- Solo uno de los vectores -vector M0, vector PIE, vector BROM, vector XINTF- debe ser habilitado a la vez.

Figura 3.40 Mapa de Memorias del DSP TMS320F2812.
Sobre el mapa de memoria, se tienen 2 tipos de memoria:
   a) Memoria dentro del chip.
   b) Memoria fuera del chip.

Debido a que se considera que la memoria que esta dentro del chip es lo suficientemente grande para el presente Trabajo Terminal, se hablará únicamente del mismo.

Ahora, dentro del cada tipo de memoria, se subdividen en dos partes:
   a) Espacio de Datos.
   b) Espacio de Programa.

El mapa de memoria contiene tres tipos de memoria de forma global:
   a) Memoria RAM
   b) Memoria ROM
   c) Memoria Flash

La memoria Flash del DSP elimina generalmente la necesidad de memoria externa así como del módulo que inicia el sistema (desde la memoria ROM), y que necesita el procesador del DSP. A pesar de las características anteriores, muchos requisitos especiales existen para poder correr una aplicación desde la memoria Flash interna del DSP TMS320F2812. Estos requisitos generalmente no son perceptibles cuando se desarrollan aplicaciones en memoria RAM, ya que el compilador del Code Composer Studio puede enmascarar problemas asociados con la inicialización de ciertos sectores que deben ser ligados a memoria.

La memoria ROM que contiene el DSP realiza varias funciones. La primera de ellas es referida a las tablas de interrupciones que utiliza el mismo DSP. La segunda de ellas es el módulo de inicio de sistema. El tercero es referido a la seguridad del DSP en hardware, es decir, se introduce una clave al DSP en una sección de Memoria que se programa una única ocasión, para tener todos los derechos de lectura del DSP.

La memoria RAM del DSP TMS320F2812 es primordial cuando se busca un sistema que funcione por sí solo, ya que debe cargar varios símbolos de configuración conforme el sistema se está iniciando. Lo anterior es cierto, ya que cuando se programa la memoria Flash existen secciones de memoria, así como variables que a pesar de encontrarse programadas en Flash, deben inicializarse desde RAM.

Finalmente, encontramos una sección de memoria dedicada a los periféricos. Lo anterior es debido al tipo de arquitectura que tiene el DSP TMS320F2812, ya que dependiendo del esquema para la comunicación serial, Texas Instruments tiene configurada a la entrada diversas formas de almacenar los datos de entrada. Lo anterior facilita considerablemente el desempeño del DSP conforme se leen datos de la comunicación serial.
3.6.3 IQmath, máquina virtual flotante

La librería IQmath para los dispositivos de Texas Instruments de la familia TMS320C28x es una colección de funciones matemáticas altamente optimizadas así como de gran precisión para los programadores de C/C++, que aparentemente ejecuta un algoritmo de punto flotante en un dispositivo de punto fijo. Estas rutinas se utilizan típicamente en aplicaciones en tiempo real que requieren de gran carga de cómputo, donde la ejecución óptima con respecto a la velocidad y a la exactitud es crítica. Utilizando estas rutinas se puede lograr una velocidad de ejecución considerablemente más rápida que su código equivalente escrito en el lenguaje C ANSÍ. Además, proporcionando las funciones de alta precisión, la librería IQmath de Texas Instruments puede acortar significativamente el tiempo de ejecución de la aplicación realizada sobre el DSP. En conclusión, IQmath no es sino una máquina virtual de punto flotante.

<table>
<thead>
<tr>
<th>Tipos de Datos IQmath: Rango y Resolución</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tipo de Dato</td>
</tr>
<tr>
<td>iq30</td>
</tr>
<tr>
<td>iq29</td>
</tr>
<tr>
<td>iq28</td>
</tr>
<tr>
<td>iq27</td>
</tr>
<tr>
<td>iq26</td>
</tr>
<tr>
<td>iq25</td>
</tr>
<tr>
<td>iq24</td>
</tr>
<tr>
<td>iq23</td>
</tr>
<tr>
<td>iq22</td>
</tr>
<tr>
<td>iq21</td>
</tr>
<tr>
<td>iq20</td>
</tr>
<tr>
<td>iq19</td>
</tr>
<tr>
<td>iq18</td>
</tr>
<tr>
<td>iq17</td>
</tr>
<tr>
<td>iq16</td>
</tr>
<tr>
<td>iq15</td>
</tr>
<tr>
<td>iq14</td>
</tr>
<tr>
<td>iq13</td>
</tr>
<tr>
<td>iq12</td>
</tr>
<tr>
<td>iq11</td>
</tr>
<tr>
<td>iq10</td>
</tr>
<tr>
<td>iq9</td>
</tr>
<tr>
<td>iq8</td>
</tr>
<tr>
<td>iq7</td>
</tr>
<tr>
<td>iq6</td>
</tr>
<tr>
<td>iq5</td>
</tr>
<tr>
<td>iq4</td>
</tr>
<tr>
<td>iq3</td>
</tr>
<tr>
<td>iq2</td>
</tr>
<tr>
<td>iq1</td>
</tr>
</tbody>
</table>

Tabla 3.2 Tipos de Datos, Resolución y Precisión IQmath
3.6.4 SCI, Interfaz de Comunicación Serial

El DSP TMS30F2812 cuenta dentro de su arquitectura con dos puertos seriales asincronos de interfaz de comunicación serial, que usualmente son conocidos como UART. Los módulos digitales SCI son utilizados para la comunicación entre el CPU del DSP con otros periféricos asincronos que hacen uso del formato de No Retorno a Cero (NRZ, Non Return to Zero). Así es entonces, que los puertos SCI del DSP TMS320F2812 pueden operar de forma independiente para una comunicación tipo half-duplex, o simultáneamente para una comunicación full-duplex.

Para asegurar la integridad de los datos, el SCI revisa los datos recibidos mediante detección, paridad, tiempo muerto, y errores de marco. Se puede programar la tasa de transmisión de bit en el código general del DSP.

3.6.5 Algoritmos en el DSP

Los algoritmos de Recuperación de Frecuencia y Fase de Portadora fueron implementados en el DSP TMS320F2812 en el entorno de desarrollo CCS. Las herramientas que ofrece Texas Instruments tienen tal nivel de integración que la programación de los algoritmos fue hecha en lenguaje C. Se hizo uso de los diagramas de flujo ya tratados aquí anteriormente para cada uno de los algoritmos obteniendo los siguientes resultados.

Se hace mención, que la programación del DSP en sí no es muy compleja, sin embargo, para poder programarlo existe un compromiso muy grande con respecto al conocimiento específico de la arquitectura del mismo DSP, cuestión que dificulta la programación del TMS320F2812.

3.7 Interfaz de comunicación serial DSP - MATLAB

Primeramente se tratarán en las siguientes secciones las características bajo las cuales tanto el DSP como MATLAB se comunican. Posteriormente se mostrarán los diagramas de flujo empleados en la programación del DSP así como el de MATLAB. Finalmente, se observarán los resultados generados del mismo.

3.7.1 Interfaz de Comunicación Serial del DSP (SCI, Serial Communications Interface)

El DSP TMS30F2812 cuenta dentro de su arquitectura con dos puertos seriales asincronos de interfaz de comunicación serial, que usualmente son conocidos como UART. Los módulos digitales SCI son utilizados para la comunicación entre el CPU del DSP con otros periféricos asincronos que hacen uso del formato de No Retorno a Cero (NRZ, Non Return to Zero). Así es entonces, que los puertos SCI del DSP TMS320F2812 pueden operar de forma independiente para una comunicación tipo half-duplex, o simultáneamente para una comunicación full-duplex.
Para asegurar la integridad de los datos, el SCI revisa los datos recibidos mediante detección, paridad, tiempo muerto, y errores de marco. Se puede programar la tasa de transmisión de bit en el código general del DSP.

Las características de transmisión tomadas dentro de la comunicación serial que se ha configurado, están las siguientes:

- Un bit de inicio.
- Ocho bits de datos.
- Un bit de paro.

Se ha decidido eliminar el bit de paridad para evitar complejidad en el tratamiento de los datos entrantes al DSP, debido a que el buffer de entrada del SCI es de 8 bits.

Es así como se vería la transmisión de información por el SCI del DSP, tanto a la entrada como a la salida:

![Diagrama de transmisión](image)

Figura 3.41 Modo de Transmisión de Datos por el SCI.

Cabe recordar que tanto el Transmisor como el Receptor deben utilizar el mismo formato para tener una comunicación clara.

Es por lo anterior, que tanto MATLAB como el DSP TMS320F2812 compranen el mismo formato anterior para su comunicación.

La información que se transmite de forma serial, no son sino los datos terminales de cada uno de los algoritmos; cabe recordar que los datos que se transmiten del DSP a MATLAB son del tipo flotante y como el DSP no es un sistema de alto nivel, dentro de la programación del mismo, se creo una rutina, la cual convierte los flotantes de 32 bits en cuatro registros de 8 bits, para después ser enviados por el puerto serial, lo anterior es por la limitante del buffer de salida con el que cuenta el SCI, ya que es de tan solo 8 bits.

El Baud Rate de 4800 fue escogido mediante pruebas realizadas de forma no tan extensa en la tarjeta de desarrollo eZdsp TMS320F2812, pero que demostró ser más confiable que las tasas superiores.
3.7.2 Diagramas de Flujo de la comunicación serial DSP - MATLAB.

Los diagramas de flujo aquí mostrados son únicamente para la transmisión serial de datos.

![Diagrama de flujo de la comunicación serial DSP](image)

Figura 3.42 Diagrama de flujo de la comunicación serial en el DSP.
Figura 3.43 Diagrama de flujo de la comunicación serial en MATLAB.
3.8 Diseño y fabricación de la tarjeta independiente

El diseño de los circuitos esquemáticos, así como para el PCB (Tarjeta electrónica impresa) se realizó con ayuda del software Protel 99SE. A continuación, se hará una descripción de los circuitos esquemáticos realizados, para poder entender el impacto que tienen sobre el funcionamiento de la tarjeta final.

3.8.1 Circuito Regulador de 5 Volts.

Este circuito regula la entrada de 12 Volts a 5 Volts, para que sea la entrada apropiada a otros circuitos que serán tratados más adelante.

3.8.2 Circuito Regulador de 3.3 Volts

Este circuito regula a 3.3 Volts. La importancia de este circuito es tal, ya que parte de la alimentación del DSP, sobre cuestiones de configuración así como de fuente y alimentación de memorias, son con este Voltaje.
3.8.3 **Circuito Acondicionador de Señal**

Este circuito acondiciona la señal que llega directamente del medio (cuando no se utiliza un Controlador Automático de Ganancia), los diferenciales de potencial a la salida son todos positivos así como el voltaje máximo es de 3.3 Volts para evitar dañar al DSP.

![Circuito Acondicionador de Señal](image)

**Figura 3.46 Circuito Acondicionador de Señal**

3.8.4 **Circuito Regulador de 1.9 Volts**

Este circuito regula a 1.9 Volts. Es necesario, ya que el CPU del DSP se alimenta con 1.9 Volts.

![Circuito Regulador de 1.9 Volts](image)

**Figura 3.47 Circuito Regulador de 1.9 Volts.**
3.8.5 Circuito de Reloj

Es el reloj con el cual el DSP se alimenta para generar la señal de reloj del DSP, cabe recalcar que el Reloj de este circuito bien podría ser el reloj de operación del DSP, pero para el presente Trabajo Terminal, el circuito de Reloj solo da la referencia de reloj, ya que internamente existe un PLL que multiplica la frecuencia del reloj para después alimentar al DSP.

![Figura 3.48 Circuito del Reloj.](image1)

### 3.8.6 Circuito MAX232

Este circuito es la interfaz física necesaria que convierte los niveles del DSP a niveles TTL para la transmisión serial, y viceversa.

![Figura 3.49 Circuito MAX232](image2)
3.8.7 Circuito del DSP

Es el circuito base que se utiliza para realizar todas las funciones asociadas al presente Trabajo Terminal. Dentro del mismo circuito existen dos etapas que alimentan al ADC del mismo DSP, estas dos etapas son meramente de configuración. En la parte superior de la imagen se observa el RESET.

El presente diagrama base fue obtenido de las hojas de especificaciones de la tarjeta de desarrollo eZdsp TMS320F2812, de la compañía "Spectrum Digital" la cual fabrica las tarjetas de desarrollo de Texas Instruments.

Finalmente se muestra el diagrama a bloques del sistema completo.
3.8.8 Tarjeta Electrónica Impresa (PCB)

Las caras de la tarjeta electrónica desarrollada, son las siguientes:

Figura 3.52 Cara superior del Circuito Impreso

Figura 3.53 Cara inferior del Circuito Impreso
Al final del proceso anterior se realizó la siguiente tarjeta electrónica:

Figura 3.54 Cara superior del Circuito.

Figura 3.55 Cara inferior del Circuito.
4. VALIDACIÓN DEL SISTEMA

La validación del sistema fue llevado a cabo con la tarjeta de desarrollo de Texas Instruments. Cabe mencionar, que los algoritmos fueron implementados en un primer momento en código flotante (operaciones de punto flotante) y finalmente el código quedo escrito en IQmath (operaciones con formato IQmath).

La validación del sistema esta hecha en dos partes, la primera de ellas, es el que la tarjeta de desarrollo funcione de forma independiente, ello implica el que la tarjeta se desconecte del cable paralelo ya que es el mismo el que lo comunica con la PC y así con Code Composer Studio, sin embargo, no habría forma de poder verificar si los algoritmos funcionan correctamente, por lo anterior, se utilizo CCS. La segunda parte es la validación del mismo pero con la comunicación serial con MATLAB, así que se verificará el desempeño del dispositivo no solo a nivel de algoritmos sino también de comunicación serial DSP - MATLAB.

![Figura 4.1 Fotografía del DSK TMS320F2812.](image)

4.1 Resultados Obtenidos del DSP TMS320F2812 con CCS.

La programación del DSP en lenguaje C no es tan alejado de la programación de una computadora al momento de utilizar variables y registros de memoria dinámica, sin embargo, cuando se desea programar alguna cuestión del DSP mismo, la programación esta orientada a programación de registros de configuración no siendo así, tan transparente como en una PC la programación normal. El DSP TMS320F2812 cuenta consigo mismo con un nuevo formato muy similar al ASCII pero que en principio es lo mismo en procesamiento, la diferencia radica en que este formato llamado IQmath realiza operaciones más pesadas en procesamiento en menor tiempo debido a que el mismo formato esta maximizado de forma conjunta con la arquitectura del DSP. IQmath es prácticamente una máquina virtual flotante, hace que el DSP TMS320F2812 que es de punto fijo, tenga niveles de respuesta tan rápidos ante operaciones de números flotantes que aplicaciones como la que se trata en este Trabajo Terminal que es de comunicaciones y donde se requiere de velocidad, no se requiera de un hardware mucho más especializado, hablando a nivel de integración de circuito integrado.
Hablando propiamente del algoritmo de "Retardo y Multiplicación sin Información", existía una gran carga computacional, debido a que de los algoritmos a implementar en el DSP TMS320F2812, realiza un mayor número de operaciones entre flotantes, así como sus variables que son de gran tamaño. Por lo anterior, existía el compromiso de tiempo-procesamiento, el cual fue llevado a cabo de forma exitosa mediante el formato IQmath. Un claro ejemplo de la capacidad de este formato es el siguiente: una multiplicación en flotante le cuesta al DSP 110 ciclos máquina poderlo resolver, con IQmath, se toman alrededor de 10 ciclos máquina. Claro esta, se paga con cierto factor de precisión, ya que IQmath realiza operaciones con múltiplos según el tipo de formato IQ que se halla configurado.

Sin más preámbulo, se muestra a continuación la gráfica de las muestras a la salida del algoritmo "Retardo y Multiplicación sin Información". Para cien símbolos, y un esquema de modulación PSK-4.

![Figura 4.2 Salida del algoritmo “Retardo – Multiplicación sin Información” en CCS.](image)

Como se puede observar, no existe gran diferencia con respecto a los resultados mostrados en el capítulo anterior (simulaciones MATLAB). Sin embargo, esta gráfica no nos permite del todo aseverar el que el algoritmo está funcionando en la forma correcta, por lo que se tomará en cuenta el diagrama a bloques mostrado en la "Solución Propuesta", que muestra a todas las etapas de recuperación de portadora, así como la del recuperador de reloj en forma continua, así que el lector del presente trabajo, podrá darse cuenta si los algoritmos realmente están funcionando de forma correcta.
Propiamente, después de la etapa de estimación del desplazamiento en frecuencia para el algoritmo de "Retardo y Multiplicación sin Información", se encuentra la etapa de recuperador de reloj, el cual en forma general, si existe un gran desplazamiento en frecuencia, aún con el algoritmo de estimación de desplazamiento en fase sin información, se observaría en pantalla un anillo de símbolos. Y en el caso contrario, en el cual no existe un desplazamiento en frecuencia que perjudique al algoritmo recuperador de reloj, el mismo podrá obtener los símbolos enviados por el transmisor, claro esta, con fase distinta. Para tener una mejor referencia con respecto al módulo recuperador de reloj, diríjase a [9].

Después de la etapa de recuperación de reloj, se encuentra el algoritmo de estimación de desplazamiento en frecuencia "Retardo y Multiplicación con Información de Temporización". Los resultados obtenidos en Code Composer Studio con respecto al algoritmo anterior son los siguientes:

![Graphical Display](image)

Figura 4.3 Salida de algoritmo “Retardo – Multiplicación con información de Temporización” en CCS.

Se puede afirmar que la señal es recuperada en frecuencia, ya que los puntos de la constelación transmitidos fueron cuatro, sin embargo, existe aún desplazamiento en fase de la portadora. El esquema propuesto como solución tiene la característica de que los algoritmos tratados en el mismo esquema son dependientes y seriados en la forma en como fueron expuestos, considerando desplazamientos en frecuencia a la entrada del receptor mayores a la tasa del símbolo, es por ello que primeramente, se realiza una estimación
La frecuencia, para que después el recuperador de reloj trabaje de forma correcta con muestras y con cierto desplazamiento en frecuencia, dando a la salida los símbolos que el mismo algoritmo trata. Posteriormente, se obtiene el desplazamiento fino en frecuencia que aún así es necesario, y finalmente se obtiene la fase correcta (cuando no existe ambigüedad) con el algoritmo de "Estimación Adelantada con QAM". Cabe recalcar que los resultados aquí mostrados, fueron obtenidos con las características anteriores del sistema.

Finalmente se muestran los resultados obtenidos del Code Composer Studio referentes al algoritmo estimador de fase "Estimación Adelantada con QAM":

El estimador de fase de la portadora no forzosamente amarra la fase en los ejes coordenados, el mismo puede ser distinto si así lo desea mediante la de-rotación deseada de la señal tratada.

Desgraciadamente, a pesar de que se muestran los cuatro puntos de la constelación PSK-4, el algoritmo no es consistente en el hecho de que amarre en la fase correcta, ya que el algoritmo de "Estimación Adelantada con QAM" presenta cierta ambigüedad cuando el desfasamiento es mayor a 45 grados, y amarra a la fase más cercana. El tema de detección
de símbolo no es tratado en el presente Trabajo Terminal, pero la propuesta que se hace, es el de utilizar una secuencia de entrenamiento para poder detectar el símbolo.

Finalmente, con la tarjeta de desarrollo eZdsp TMS320F2812 junto con Code Composer Studio se puede decir que el nivel de operatividad es del 100%, y es claro, ya que la tarjeta al estar funcionando a través de la administración de CCS, todas las circunstancias que se verían en un sistema completamente independiente se nulifican gracias a la interacción continua de DSP - PC (CSS).

4.2 Resultados Obtenidos del DSP TMS320F2812 con MATLAB.

Para el caso en el cual la tarjeta de desarrollo funciona de forma independiente (sin cable paralelo), se realizó una interfaz serial con el circuito MAX-232 (vea la descripción de éste circuito en el capítulo anterior) de Texas Instruments para poder enviar los datos mediante el puerto SCIB del DSP hacia una computadora que corre un programa creado en MATLAB; lo anterior es debido a que el DSP en modo independiente no puede comunicarse con algún medio directamente para poder ver los resultados deseados, es por lo anterior que se desarrolló en el Trabajo Terminal esta interfaz.

En la interfaz, MATLAB es el que realiza una función "amo" y el DSP realiza la función "esclavo". El DSP espera a que MATLAB le envíe el byte de configuración para que ciertos algoritmos funcionen durante el proceso. Durante la transmisión de datos, el DSP envía cuatro datos enteros que fueron obtenidos directamente de un dato flotante, es así como se envían los cuatro bytes, para después tener que esperar el acuse.

La validación del sistema en este punto, será con respecto a las ocasiones cuando la comunicación serial llega a bloquearse; lo anterior es sencillo, ya que no se cuenta con un protocolo avanzado sino con uno simple llamado “para y espera” (stop and wait, como lo refiere la bibliografía). Se consideró una transmisión de 100 símbolos en condiciones de laboratorio. Además se realizaron 50 pruebas sobre el presente sistema para cada uno de los casos. Considere que los algoritmos corriendo internamente en el DSP no llegan a bloquearse ni muchos menos dejan al DSP en estado de peligro, los errores aquí generados son meramente del momento en el cual se transmiten los datos por el puerto serie.

<table>
<thead>
<tr>
<th>Algoritmos</th>
<th>RR &lt; 5</th>
<th>5 ≤ RR &lt; 10</th>
<th>10 ≤ RR &lt; 15</th>
<th>15 ≤ RR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retardo – Multiplicación Sin I.</td>
<td>24</td>
<td>14</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>Oerder Interpolation</td>
<td>8</td>
<td>21</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>Retardo – Multiplicación con I. de T.</td>
<td>7</td>
<td>18</td>
<td>18</td>
<td>7</td>
</tr>
<tr>
<td>Estimación Adelantada con QAM</td>
<td>9</td>
<td>16</td>
<td>16</td>
<td>9</td>
</tr>
<tr>
<td>Todos los anteriores</td>
<td>7</td>
<td>19</td>
<td>20</td>
<td>4</td>
</tr>
</tbody>
</table>

Tabla 4.1 Tabla de funcionamiento de la interfaz serial DSP – MATLAB.
Lo anterior se comprende, ya que para el algoritmo "Retardo - Multiplicación sin Información" tiene a su salida 400 números reales y 400 números complejos, es por lo anterior que existe una mayor probabilidad de error en la transmisión serial. Para todos los demás casos, se ve una similitud y ésta se debe precisamente a que a la salida de cualquier otro algoritmo se tienen 100 símbolos de salida para los reales así como de 100 para los complejos.

Como resultado de la interfaz gráfica podemos observar a la siguiente figura, la cual es la interfaz hecha para la transmisión serial.

Figura 4.5 Interfaz Serial de la PC con el programa MATLAB.
5. CONCLUSIONES

5.1 Conclusiones


Desafortunadamente el objetivo general no ha sido alcanzado debido a diversas circunstancias, las cuales no es necesario mencionarlas. Se han implementado diversos algoritmos de Recuperación de Portadora en el DSP, así como se diseñó y construyó el módulo de Recuperación de Portadora, pero el mismo no funcionó correctamente en la presente fecha.

Con respecto a los objetivos particulares, la comprensión del Problema de Portadora es con sus limitaciones, ya que en un aspecto matemático los análisis son realmente complejos y quedan fuera de las herramientas con las que cuenta el estudiante. Por otro lado, la comprensión física del problema ha sido revisado en la misma realidad, es decir, con ayuda de los asesores de este trabajo, se observaron fenómenos que ocurren cuando no existe una Recuperación de Portadora, así como del efecto del mismo a nivel teórico y como ejemplo tenemos al efecto Doppler, ya antes descrito en la Introducción del presente Trabajo Terminal.

Se realizó una revisión de algoritmos y es de ahí donde el presente trabajo tiene sus antecedentes, ya que el Problema de Portadora no es un tema que este escrito en cualquier libro, sino de algunos muy especializados sobre el tema de Sincronización así como de la Recuperación de Portadora, para nuestro caso se utilizó la referencia [2], así como de los artículos publicados por las personas relacionadas a los distintos algoritmos tomados finalmente. Es de estas fuentes donde se observó como los Recuperadores de Portadora se clasifican, teniendo desde los ideales (que tienen conocimiento de sincronización, están asistidos por datos, así como suponen canales ideales, etc.) hasta los que pueden ser utilizados en la vida real, debido a las características que toman en cuenta. La recuperación de Portadora no solamente esta dirigida a la estimación y corrección en frecuencia sino también a la estimación y corrección de fase, para poder obtener una Recuperación de Portadora por así decirlo, más completa; ya que durante el presente escrito se han mencionado las características necesarias para que los algoritmos tengan un buen desempeño.

Con respecto a la elección de los algoritmos, fue realizada por los asesores, pero que a final de cuentas es comprendido el porque los algoritmos aquí presentados tienen las características para poder funcionar de forma correcta, por ejemplo, el algoritmo de "Retardo - Multiplicación sin información" aparentemente funciona para distintos esquemas de modulación, no así el "Retardo - Multiplicación con información de • Temporización" ni tampoco el de "Estimación Adelantada con QAM", que están limitados a trabajar con esquemas de modulación QAM-4. En dado caso que se desease utilizar otro esquema de modulación, dos de tres algoritmos son inútiles de forma directa, así como el otro debería ser analizado para poder verificar si es compatible con el formato de modulación.
Las simulaciones reportadas en el presente Trabajo Terminal han sido descritas en el capítulo 3, de donde podemos concluir lo siguiente:

- El algoritmo de "Retardo y Multiplicación sin Información" tiene un mejor desempeño conforme el exceso de ancho de banda es mayor.
- El algoritmo de "Retardo y Multiplicación con información de Temporización" tiene un mejor desempeño conforme el intervalo de observación (número de datos a la entrada) crece.
- El algoritmo "Estimación Adelantada con QAM" tiene un buen desempeño cuando el desplazamiento en fase no es mayor a 45°, ya que si se llega a exceder de éste límite, se amarra la fase en el punto más cercano de amarre, provocando así una ambigüedad del algoritmo para desplazamientos más grandes.

Con respecto al aprendizaje del funcionamiento del DSP TMS320F2812 de Texas Instruments me deja muy en claro que la programación conforme cada vez es más elevada no es del todo más sencilla, hay ocasiones en las cuales se deben utilizar primitivas no tan usadas pero que si son necesarias para que el sistema funcione de forma correcta. Queda también muy claro el que la programación de un dispositivo como el DSP, requiere de un compromiso muy grande para la comprensión así como de la memorización de la arquitectura del mismo DSP, para evitar fallar en la configuración del mismo. Lo más significativo que me llevo de estos dispositivos es la exactitud con la cual ejecutan cualquier tarea, así como de la velocidad de procesamiento de datos con la que llegan a trabajar dispositivos como estos.

El DSP TMS320F2812 no se programa por sí solo, Texas Instruments provee una herramienta para poder programar, simular en tiempo real, así como muchas otras tareas; llamada Code Composer Studio. CCS es un entorno de desarrollo muy fuerte, a tal grado que no hay función que sea sobrante así como no hay función que sea faltante del mismo entorno. Se cuenta con análisis den tiempo real, con procesamiento mucho más rápido de lo normal (formato IQ), gráficas en tiempo, frecuencia, etc.

Finalmente se debió haber creado una tarjeta sobre la cual estuviesen montados los distintos algoritmos tratados en este escrito, desgraciadamente, no funcionó el circuito dado, debido a razones que hasta la fecha no se han podido determinar. Todo el diseño de la tarjeta fue desarrollado en un programa llamado PROTEL 99SE, el cual permite generar un circuito esquemático para después exportarlo a un archivo de Tarjeta Electrónica Impresa y así fabricar la tarjeta diseñada.

5.2 Mejoras

Las mejoras que son visibles, es la realización de la tarjeta independiente. Así como la implementación de otros algoritmos de Recuperación de Portadora para distintos esquemas de modulación utilizados. De igual forma, para poder verificar la señal a la salida del algoritmo que recupera la fase debería haber un método de detección para verificar que los datos en fase sean los correctos y en otro caso, sería el desarrollar algún algoritmo por medio de secuencia de entrenamiento para poder tener una buena decisión de símbolo.
REFERENCIAS

BIBLIOGRAFÍA

- "TMS320C28x DSP CPU and Instruction Set Reference Guide", Texas Instruments, Marzo 2004, Revisión D.
- "Running an Application from Internal Flash Memory on the TMS320F281x DSP", Texas Instruments, Septiembre 2004, Revisión D.
- "IQmath Library; A virtual floating point engine", Texas Instruments, Junio 2002, Versión 1.4.1.
Amplitud: El tamaño o magnitud de una señal
Ancho de Banda: Diferencia entre las frecuencia límite de un espectro de frecuencia continuo.
Atenuación: Disminución en magnitud de la señal durante su transmisión.
Banda Base: Transmisión de señales sin modulación. En una red de banda base, las señales digitales (uno y ceros) se insertan directamente en el cable como pulsos de tensión. Todo el espectro del cable es utilizado por la señal.
Baudio: Unidad de velocidad de la señal, dada por el número de valores discretos o eventos de una señal por segundo, o la inversa del tiempo de duración del elemento de señal más corto.
Bit de Paridad: Un bit de comprobación añadido a un conjunto de dígitos binarios para hacer al conjunto de todos los dígitos binarios de valor uno, incluyendo el bit de comprobación, siempre par o impar.
Bits de relleno: La inserción de bits extra en una cadena de datos para evitar la aparición de secuencias de control no deseadas.
Decibelio: Medida de la intensidad relativa de dos señales. El número de decibelios es 10 veces el logaritmo del cociente de la potencia de dos señales, ó 20 veces el logaritmo del cociente de tensión de dos señales.
Fase: Posición relativa en el tiempo dentro de un periodo individual de la señal.
Frecuencia: Velocidad de oscilación de la señal en Hertz.
Medio de Transmisión: Camino físico entre transmisores y receptores en un sistema de comunicación.
Modulación: Proceso, o resultado del proceso, de variación de algún parámetro de una señal, llamada portadora, de acuerdo con una señal mensaje.
Modulación de Fase: Modulación en la que el ángulo de fase de una portadora es el parámetro que se varía.
Modulación en Amplitud: Una forma de modulación en la que la amplitud de la onda portadora varía de acuerdo con alguna característica de la señal modulante.
Periodo: Valor absoluto del mínimo intervalo tras el que se obtienen los mismos valores de una onda periódica.
Portadora: Frecuencia continua capaz de ser modulada o readaptada por una segunda señal (portadora de información).
Protocolo: Conjunto de reglas que gobiernan la operación de unidades funcionales para llevar a cabo la comunicación.
Ruido: Señales no deseadas que se combinan con la señal de transmisión o de recepción y que por tanto la distorsionan.
Señal analógica: Onda electromagnética que varía continuamente y se puede propagar por medio diversos.
Señal digital: Una señal discreta o discontinua.